

## IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of: Pekka Lehtinen

Serial No.:

Group No.:

Filed: Herewith

Examiner:

For: A METHOD AND A SYSTEM FOR EXECUTING APPLICATION SESSIONS  
IN AN ELECTRONIC DEVICE, AND AN ELECTRONIC DEVICECommissioner for Patents  
Alexandria, VA 22313-1450

## TRANSMITTAL OF CERTIFIED COPY

Attached please find the certified copy of the foreign application from which priority is claimed for this case, along with the certified translation of Tuulikki Tulivirta, Certified Translator:

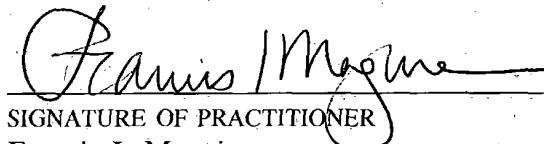
Country: Finland  
Application Number: 20021204  
Filing Date: June 20, 2002

WARNING: "When a document that is required by statute to be certified must be filed, a copy, including a photocopy or facsimile transmission of the certification is not acceptable." 37 C.F.R. 1.4(f) (emphasis added).

Reg. No.: 31,391

Tel. No.: (203) 261-1234

Customer No. 004955



SIGNATURE OF PRACTITIONER

Francis J. Maguire

Ware, Fressola, Van Der Sluys & Adolphson LLP

(type or print name of practitioner)

755 Main Street, P.O. Box 224

P.O. Address

Monroe, Connecticut 06468

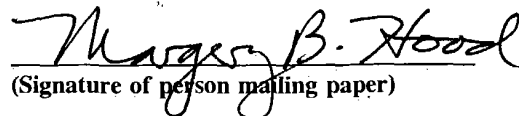
NOTE: The claim to priority need be in no special form and may be made by the attorney or agent, if the foreign application is referred to in the oath or declaration, as required by § 1.63.

## CERTIFICATE OF MAILING (37 CFR 1.10)

I hereby certify that this correspondence is, on the date shown below, is being deposited with the United States Postal Service on the date shown below is being deposited with the United States Postal Service with sufficient postage as Express Mail No. EV252882394US in an envelope addressed to the: Mail Stop Patent Application, U.S. Patent and Trademark Office, P.O. Box 1450, Alexandria, VA 22313-1450.

Date: June 20, 2003Margery B. Hood

(Type or print name of person mailing paper)



(Signature of person mailing paper)

(Transmittal of Certified Copy [5-4])

PATENTTI- JA REKISTERIHALLITUS  
NATIONAL BOARD OF PATENTS AND REGISTRATION

Helsinki 28.4.2003

ETUOIKEUSTODISTUS  
PRIORITY DOCUMENT



Hakija  
Applicant

Nokia Corporation  
Helsinki

Patenttihakemus nro  
Patent application no

20021204

Tekemispäivä  
Filing date

20.06.2002

Kansainvälinen luokka  
International class

G06F

Keksinnön nimitys  
Title of invention

**"Menetelmä ja järjestelmä sovellusistuntojen suorittamiseksi  
elektroniikkalaitteessa, ja elektroniikkalaite"**

Täten todistetaan, että oheiset asiakirjat ovat tarkkoja jäljennöksiä  
Patentti- ja rekisterihallitukselle alkuaan annetuista selityksestä,  
patenttivaatimuksista, tiivistelmästä ja piirustuksista.

This is to certify that the annexed documents are true copies of the  
description, claims, abstract and drawings originally filed with the  
Finnish Patent Office.

  
Pirjo Kaila  
Tutkimussihteeri

Maksu 50 €  
Fee 50 EUR

Maksu perustuu kauppa- ja teollisuusministeriön antamaan asetukseen 1027/2001 Patentti- ja rekisterihallituksen maksullisista suoritteista muutoksineen.

The fee is based on the Decree with amendments of the Ministry of Trade and Industry No. 1027/2001 concerning the chargeable services of the National Board of Patents and Registration of Finland.

Osoite:	Arkadiankatu 6 A	Puhelin:	09 6939 500	Telefax:	09 6939 5328
	P.O.Box 1160	Telephone:	+ 358 9 6939 500	Telefax:	+ 358 9 6939 5328
	FTN-00101 Helsinki, FINLAND				

Menetelmä ja järjestelmä sovellusistuntojen suorittamiseksi elektroniikkalaitteessa, ja elektroniikkalaitte

- 5 Nyt esillä oleva keksintö kohdistuu järjestelmään, joka käsittää välineet sovellusistuntojen suorittamiseksi elektroniikkalaitteessa, jossa on yksi tai useampi suoritin, ja välineet resurssivarausten sekä olennaisesti samanaikaisesti suoritettavana olevien sovellusistuntojen suorituksen vuorottelemiseksi. Keksintö kohdistuu lisäksi menetelmään sovellusistuntojen suorittamiseksi elektroniikkalaitteessa, jossa toimittaa ohja-
- 10 taan yhdellä tai useammalla suorittimella, ja jossa menetelmässä resurssivarauksia sekä olennaisesti samanaikaisesti suoritettavana olevien sovellusistuntojen suoritusta vuorotellaan. Keksintö kohdistuu vielä elektroniikkalaitteeseen, joka käsittää välineet sovellusistuntojen suorittamiseksi, yhden tai useamman suorittimen, ja välineet resurssivarausten sekä olennaisesti samanaikaisesti suoritettavana olevien sovellusistuntojen suorituksen vuorottelemiseksi.

- 20 Langattomien viestintälaitteiden ominaisuuksien lisääntyessä on tarvo kehittää järjestelmiä, jolla näitä ominaisuuksia voidaan käyttää järkevästi ja monipuolisesti hyväksi. Nykyisissä langattomissa viestintälaitteissa on mahdollista jopa suorittaa useita tarkoituksellään erilaisia sovellusistuntoja (lyh. istuntoja, engl. application sessions, sessions), kuten kalenteri, ajastin, puhelinluettelo, muistio, yms. Usein voi olla tarpeen suorittaa useita tällaisia istuntoja olennaisesti samanaikaisesti, jolloin ongelmia voi tulla mm. silloin, kun eri istunnot tarvitsisivat samanaikaisesti samoja langattoman viestintälaitteen omia tai yhteyksien kautta käytettävissä olevia resursseja käyttöönsä. Tunnetaan esim. tietojenkäsittelylaitteisiin tarkoitettuja käyttöjärjestelmiä, kuten UNIX®,
- 25 joissa on toteutettu moniajomenetelmä, eli useiden istuntojen olennaisesti samanaikainen suorittaminen. Tällainen moniajomenetelmä on toteutettu esimerkiksi siten, että kukin istunto on toteutettu yhden ohjelmajärjestelmän muodossa ja käyttöjärjestelmä antaa kullekin ohjelmajärjestelmälle suoritusaikaa tietyssä järjestyksessä. Tällöin yhden istunnon ollessa suorituksessa muut istunnot ovat odotustilassa. Istuntoja ei siis käytännössä suoriteta samanaikaisesti, vaan peräkkäin. Kuitenkin kullekin istunnolle kulloinkin annettava suoritus aika on niin lyhyt, että
- 30
- 35

## 2

syntyy vaikutelma siitä, että kaikkia istuntoja suoritetaan samanaikaisesti. Tällaisessa järjestelyssä mikään istunto ei normaalisti joudu odottamaan niin pitkää aikaa, että istunnon suoritus näyttäisi keskeytyneen. Kuitenkin tilanteessa, jossa samanaikaisesti suoritettavana olevien istuntojen määrä kasvaa, jää kullekin istunnolle entistä vähemmän suoritusaikaa, mikä hidastaa kaikkien istuntojen toimintaa. Joissakin järjestelmissä on mahdollista jaotella istunnot tarkuysjärjestykseen (priorisointi), jolloin istunnoille annetaan suoritusaikaa niiden tärkeysjärjestyksen mukaisesti. Tässä järjestelyssä vähemmän tärkeiden istuntojen suoritus saattaa hidastua merkittävästi.

Ongelmana tunnetun tekniikan mukaisissa moniajojärjestelmissä on ollut myös se, että jos jokin istunto tarvitsee jotakin laiteresurssia ja/tai käyttöjärjestelmäresurssia, varaa tämä istunto resurssin niin pitkäksi ajaksi kuin resurssin käyttö on tarpeen. Tällöin muut sellaiset istunnot, jotka tarvitsisivat samaa resurssia, joutuvat odottamaan kyseisen resurssin vapautumista, koska ei ole yleensä käytettävissä menetelmää, jolla resurssi voitaisiin siirtää yhdellä istunnolta toiselle siten, että tämä resurssin käytettävyytilanne olisi kullakin istunnolla tiedossa. Tällaisia tilanteita varlen on kehitelty joitakin ratkaisuja, joissa käyttöjärjestelmä käyttää opastimia (semaphore) tai vastaavia estääkseen näiden avulla varalluna olevan resurssin käytön toisilla silä pyytäviltä istunnoilta.

Useiden langattomissa viestintälaitteissa suoritettavien istuntojen suorituksen tulisi olla mahdollisimman tosiaikaista mm. langattoman viestintälaitteen käyttömukavuuden kannalta. Esimerkiksi puhelua aloitettaessa puhelinmuistioistunto tulisi suorittaa niin nopeasti kuin mahdollista, jotta langattoman viestintälaitteen käyttäjä löytäisi etsimänsä puhelinnumeron ilman haitallista viivettä. Tällaisen tosiaikaisuuden toteuttaminen voi kuitenkin olla hankalaa erityisesti silloin, kun samanaikaisesti suoritettavana olevia istuntoja on langattomassa viestintälaitteessa käynnissä runsaasti. Yksi tunnettu ratkaisu tosiaikaisuuden ja moniajon vaatimusten täyttämiseksi on useamman suorittimen käyttö samassa laitteessa. Tällöin eri istuntoja voidaan suorittaa eri suorittimilla. Tässä on kuitenkin mm. se epäkohta, että suorittimien määrän lisääntyessä myös laitteen tehonkulutus kasvaa, mitä tulisi välttää erityisesti kannettavissa laitteissa. Lisäksi usean prosessorin järjestelmis-

## 3

sä tarvitaan järjestelyjä, joilla eri istuntoja jaetaan eri suorittimille suoritettavaksi.

- 5 Sulautettuihin järjestelmiin on kehitetty moniajojärjestelmiä, joissa eri istuntojen suoritusta voidaan ajastaa ja antaa kullekin istunnolle suoritusaikaa tarvittaessa. Tällaiset sulautetut järjestelmät on kuitenkin suunniteltu vain tiettyihin käyttötarkoituksiin, jolloin voidaan etukäteen selvittää suurelta osin se, mitä istuntoja eri tilanteissa suoritetaan ja mitä resursseja ne kulloinkin tarvitsevat. Tällöin istuntojen ajastaminen voidaan suorittaa etukäteen. Nämä järjestelmät eivät myöskään sisällä käyttöjärjestelmää. Tällaiset järjestelmät eivät sovellu käytettäväksi langattomissa viestintälaitteissa, joissa ei voida etukäteen selvittää läheskään kaikkia eri käyttötilanteita ja niissä tarvittavia resursseja.
- 10
- 15 Eräinä merkittävinä ongelmina moniajojärjestelmissä on resurssien hallinta ja ajoitus sekä sovellusistuntojen hallinta ja ajoitus, loisin sa-  
noen suorittimen ja muiden eri tyyppisten resurssien varausten synkronoin-  
ti eri istuntojen tarpeita vastaavasti. Näitä ongelmia ei kuitenkaan  
voida ratkaista erillisinä, jolloin molempiin ongelmiin sopivien ratkaisui-  
den löytäminen on vaikeaa.
- 20

- Nyt esillä olevan keksinnön eräänä tarkoituksena on aikaansaada pa-  
rannettu järjestely erityisesti langattomia viestintälaitteita varten mah-  
dollisimman tosiaikaisen moniajojärjestelmän toteuttamiseksi. Keksintö
- 25 perustuu siihen ajatukseen, että muodostetaan istuntojen suoritusym-  
päristö, jossa on erilaisia toiminnallisia kokonaisuuksia istuntojen suo-  
rituksen ja resurssien käytön ohjaamiseksi. Keskeisenä näistä toimin-  
nallisista kokonaisuuksista ovat sovellusistuntojen ja resurssien oh-  
jaustoiminnot, jotka jakautuvat mm. sovellusistuntojen hallinta- ja ajoi-  
tuslohkoon ja resurssien hallinta- ja varauslohkoon. Suoritettava istunto
- 30 on jaettu peräkkäisiin ja rinnakkaisiin tehtäviin (activity), jotka on to-  
teutettu vastaavasti sovelluslohkoina (Application Block). Sovellusis-  
tuntojen hallinta- ja ajoituslohko (Application and Scheduling Manager)  
vastaa istuntojen kulloistenkin tehtävien ajoituksesta (käynnistämisestä)  
35 mm. käytettävissä olevien resurssien perusteella. Resurssien hal-  
linta- ja varauslohko (Resource Allocation Manager) vastaa kaikkiin eri  
resurssityyppeihin ja resursseihin liittyvistä keskitetyistä hallinta- ja va-

- raustoiminnoista mm. pitämällä kirjaa varattuna olevista resursseista ja ilmoittamalla niistä istuntojen hallinta- ja ajoituslohkulle. Eräänä keskeisenä piirteenä nyt esillä olevan keksinnön mukaisen järjestelmän toiminnassa on sovellusistuntojen hallinta- ja ajoituslohkon ja resurssien
- 5 hallinta- ja varauslohkon keskinäinen ajoitus, joka on toteutettu siten, että resurssien hallinta- ja varauslohkon suorituksen jälkeen, jolloin ainakin joidenkin resurssien varaustilanne on stabiili ja mahdollisimman ajantasainen, suoritetaan olennaisesti välittömästi sovellusistuntojen hallinta- ja ajoituslohkon toimintoja. Muita toiminnallisia kokonaisuuksia
- 10 ovat mm. resurssien välittimet (Resource Handler) kutakin resurssityyppiä varten sekä sovelluslohkosäiliöt (Application Block Container). Resurssien välittimet ylläpitävät kyseisen resurssityypin varauksista luetteloa omassa resurssivaraustaulussaan (Resource Instance Table). Kukin ohjelmalohkosäiliö pitää sisällään edullisesti
- 15 sellaisia yhden tai useamman istunnon tarvitsemia sovelluslohkoja, joiden suoritus on tavanomaisesti peräkkäistä ilman tarvetta samanaikaiseen suoritukseen. Resurssivälittimet voidaan edullisesti toteuttaa ohjelmajärjestelmänä, joiden prioriteetit ovat korkeampia kuin resurssien hallinta- ja varauslohkon prioriteetti, jolloin resurssien hallinta- ja va-
- 20 rauslohko saa käyttöjärjestelmältä suorittimeen tilanteessa, jolloin resurssivälittimet ovat käsitelleet sovellusten lähettämät sanomat sekä resursseilta päin tulleet sanomat ja resurssien varaustilanteen voidaan todeta olevan hetkellisesti yksikäsitteisesti tunnettu ainakin joidenkin resurssien osalta. Keksinnön eräänä toiminnallisena piirteenä on vielä
- 25 istunnonohjausprotokolla (Session Control Protocol), jonka avulla huolehditaan mm. sanomien välityksestä järjestelmän eri osien välillä.

- Täsmällisemmin ilmaistuna nyt esillä olevan keksinnön mukaiselle järjestelmälle on pääasiassa tunnusomaista se, että suoritettava sovel-
- 30 lusistunto käsittää yhden tai useamman sovelluslohkon yhdessä tai useammassa sovelluslohkosäiliössä, ja mainituille sovelluslohkoille on määritetty suoritusjärjestys, että järjestelmä käsittää resurssityyppi-kohtaisla resurssivälittimä resurssien varaamiseksi sovellusistunnon käyttöön, resurssien hallinta- ja varausvälineet resurssien varaustilan-
- 35 teen tutkimiseksi ja tallentamiseksi, sovellusistuntojen hallinta- ja ajoitusvälineet ainakin mainitun varaustilanteen perusteella seuraavaksi suoritusvuoroon tulevan sovellusistunnon valitsemiseksi, suoritusväli-

## 5

- neet valitussa sovellusistunnossa suoritusvuorossa olevan sovelluslohkon valitsemiseksi ja suorittamiseksi, ja järjestelmään on muodostettu mainittujen resurssivälittimien, resurssien hallinta- ja varausvälineiden, sovellusistuntojen hallinta- ja ajoitusvälineiden ja suoritusvälineiden välinen yhteyskäytäntö suoritusjärjestyksen määrittämiseksi ja tiedon siirron toteuttamiseksi mainittujen varausvälineiden, resurssien hallinta- ja varausvälineiden, sovellusistuntojen hallinta- ja ajoitusvälineiden ja suoritusvälineiden välillä.
- 5
- 10 Nyt esillä olevan keksinnön mukaisella menetelmälle on pääasiassa tunnusomaista se, että suoritettava sovellusistunto käsittää yhden tai useamman sovelluslohkon yhdessä tai useammassa sovelluslohosäiliössä, ja mainituille sovelluslohkoille määritetään suoritusjärjestys, että menetelmässä suoritetaan ainakin seuraavia vaiheita:
- 15
- resurssien hallinta- ja varausvaihe resurssien varaamiseksi sovellusistunnon käyttöön,
  - tutkimis- ja tallennusvaihe resurssien varaustilanteen tutkimiseksi ja tallentamiseksi,
  - valintavaihe seuraavaksi suoritusvuoroon tulevan sovellusistunnon valitsemiseksi ainakin mainitun varaustilanteen perusteella,
- 20
- suoritusvaihe valitussa sovellusistunnossa suoritusvuorossa olevan sovelluslohkon valitsemiseksi ja suorittamiseksi,
- jolloin menetelmässä käytetään mainittujen resurssien hallinta- ja varausvaiheen, tutkimis- ja tallennusvaiheen, valintavaiheen, ja suoritusvaiheen välistä yhteyskäytäntöä suoritusjärjestyksen määrittämiseksi ja tarvittaessa tiedon siirtämiseksi mainittujen resurssien hallinta- ja varausvaiheen, tutkimis- ja tallennusvaiheen, valintavaiheen, ja suoritusvaiheen välillä.
- 25
- 30 Nyt esillä olevan keksinnön mukaiselle elektroniikkalaitteelle on vielä pääasiassa tunnusomaista se, että suoritettava sovellusistunto käsittää yhden tai useamman sovelluslohkon yhdessä tai useammassa sovelluslohosäiliössä, ja mainituille sovelluslohkoille on määritetty suoritusjärjestys, että elektroniikkalaite käsittää resurssityyppikohtaisia resurssivälittimiä resurssien varaamiseksi sovellusistunnon käyttöön, resurssien hallinta- ja varausvälineet resurssien varaustilanteen tutkimiseksi ja tallentamiseksi, sovellusistuntojen hallinta- ja ajoitusvälineet
- 35

## 6

ainakin mainitun varaustilanteen perusteella seuraavaksi suoritusvuoroon tulevan sovellusistunnon valitsemiseksi, suoritusvälineet valitussa sovellusistunnossa suoritusvuorossa olevan sovelluslohkon valitsemiseksi ja suorittamiseksi, ja elektroniikkalaitteeseen on muodostettu vä-

5 lineet mainittujen resurssivälittimien, resurssien hallinta- ja varausvälineiden, sovellusistuntojen hallinta- ja ajoitusvälineiden ja suoritusvälineiden välinen yhteyskäytäntö suoritusjärjestyksen määrittämiseksi ja tiedonsiirron toteuttamiseksi mainittujen varausvälineiden, resurssien hallinta- ja varausvälineiden, sovellusistuntojen hallinta- ja ajoitusvälineiden ja suoritusvälineiden välillä.

10

Nyt esillä olevalla keksinnöllä saavutetaan merkittäviä etuja tunnetun tekniikan mukaisiin ratkaisuihin verrattuna. Keksinnön mukaisella menetelmällä yksi istunto ei välttämättä varaa resursseja itselleen kokonaan eikä välttämättä koko istunnon ajaksi, jolloin resursseja voidaan olennaisesti samanaikaisesti jakaa useammalle kuin yhdelle istunnolle ja synkronoida istunnon samanaikaisesti tarvitsemien resurssien varausajat mahdollisimman edullisella tavalla. Resurssien sekä suorittimen hyödyntäminen istuntojen suorituksessa saadaan tehokkaasti toteutettua keksinnön mukaisessa järjestelmässä olellaen, että sovellusistuntojen hallinta- ja ajoituslohkoon sisällytetään riittävä ohjausälykyys. Keksinnön mukaisella järjestelyllä voidaan ylikuormitustilanteet (overload) hallita ja välttää lukittumistilanteiden (deadlock) syntymistä. Ylikuormitustilanteissa voidaan tärkeysjärjestyksen mukaisesti käynnissä olevia sovellusistuntoja keskeyttää ja viivyttää uusien sovellusistuntojen aloittamista. Tämä koskee muidenkin jaettujen resurssien ylikuormitustilanteita, eikä vain suorittimen ylikuormitustilanteita. Keksinnön mukaisella istunnonohjausprotokollalla ja eri toimintoja suorittavien ohjelmaprosessien priorisointiperiaatteella kyetään toimintojen ajoitus hallitsemaan myös suorittimen ylikuormitustilanteissa, jolloin useilla ohjelmaprosesseilla voi olla samanaikaisesti useita käsittelemättömiä sanomia. Samanaikaisesti käsittelemättömänä olevien sanomien käsittelyjärjestys määräytyy sovellusistunnon suorittamiseen osallistuvien ohjelmaprosessien priorisointiperiaatteen mukaisesti käyttöjärjestelmän normaalin toimintatavan puitteissa. Lisäksi keksinnön mukaisella järjestelyllä saavutetaan se etu, että langattoman viestintälaitteen tehon-

15

20

25

30

35

kulutusta voidaan pienentää tilanteissa, joissa kuormitus tilanne on niin



5  
10  
15  
20  
25

pieni, että suoritin voi toimia vajaateholla. Sovellusistuntojen hallinta- ja ajoituslohko tekee mahdolliseksi tehonkulutuksen kannalta erilaisten kuomitustilanteiden havaitsemisen ohjausta varten. Keksinnön mukainen ohjelmistoarkkitehtuuri tarjoaa yhtenäisen ohjelmistoympäristön, jolloin sovelluslohkojen laatiminen langatonta viestintälaitetta varten on mahdollisimman yhdenmukaista riippumatta siitä, kuinka monimutkaisesta tai yksinkertaisesta istunnosta ja siihen kuuluvasta tehtävästä on kyse. Istuntojen lohkorakenne mahdollistaa myös sen, että istuntojen suoritukseen tarvittavan ohjelmointityö voidaan toteuttaa helposti jakamalla istunto erillisiin tehtäviin, joista kukin muodostaa yhden sovelluslohkon. Lohkosäiliöiden tarjoamaan sovelluskehikseen (software design pattern) voidaan sisällyttää kaikkien sovellusistuntojen ohjaus-  
sanomien lähetys ja vastaanotto ilman, että sovelluksen suunnittelijan tarvitsee olla tietoinen istunnonohjausprotokollan, sovellusistuntojen hallinta- ja ajoituslohkon sekä resurssien hallinta- ja varauslohkon olemassaolosta. Keksinnön mukaisen ohjelmistoarkkitehtuurin sovelluslohkoihin perustuvassa ohjelmistokehityksessä kyetään hyödyntämään tehokkaita ohjelmistonkehitystyökaluja ja uudelleen käyttämään tehokkaasti sekä sovelluslohkota että niitä istuntoprofiiliksi yhdistävää toiminnallista rakennetta. Keksinnön mukaisessa arkkitehtuurissa sovellusistuntojen hallinta- ja ajoituslohko huolehtii kunkin sovelluslohkon asianmukaisesta käynnistämisestä.

25

Keksintöä selostetaan seuraavassa tarkemmin viitaten samalla oheisiin piirustuksiin, joissa

kuva 1 esittää keksinnön erään edullisen suoritusmuodon mukaista langatonta viestintälaitetta pelkistettynä lohkokaaavana,

30 kuva 2 esittää keksinnön erään edullisen suoritusmuodon mukaista järjestelmää pelkistetyt,

35 kuva 3 esittää keksinnön erään edullisen suoritusmuodon mukaisessa järjestelmässä toteutetun sovelluslohkosäiliön rakennetta ja toimintaa pelkistetyt, ja

kuva 4 esittää keksinnön mukaisen menetelmän toimintaa oräässä  
esimerkki-istunnon tilanteessa.

5 Seuraavassa keksintöä kuvataan siltä osin, että keksinnön toteutuk-  
sessa tarvittavat toiminnot käyvät ilmi. On kuitenkin selvää, että vaikka  
jatkossa järjestelmän eri toimintoja on kuvattu lohkorakenteisesti jaet-  
tuna, on selvää, että useiden eri toiminnallisuuksien toteuttamiseen  
osallistuvat useat lohkot istunnonohjausprotokollan avulla tapahtu-  
vassa yhteistyössä.

10

Kuvassa 1 on esitetty keksinnön erään edullisen suoritusmuodon mu-  
kalnen langaton viestintälaite 1 pelkistettynä lohkoaviona. Langaton  
viestintälaite 1 käsittää ohjauslohkon 5, jossa on edullisesti suoritin 2  
(MCU) langattoman viestintälaitteen 1 toimintojen ohjaamiseksi, käyt-  
15 töjärjestelmätoimintojen suorittamiseksi jne. Lisäksi langattoman vies-  
tintälaitteen 1 ohjauslohkossa 5 voi olla digitaalinen signaalinkäsittely-  
yksikkö 3 (DSP) signaalinkäsittelytoimintoja varten. Suorittimelle 2 ja  
digitaaliseen signaalinkäsittely-yksikölle 3 on järjestetty myös muistia 4  
käyttöjärjestelmän, ohjelmakoodin, toiminnassa tarvittavien tietojen  
20 yms. tallentamista varten. Langattomassa viestintälaitteessa 1 ovat  
myös tiedonsiirtovälineet 6, kuten välineet matkapuhelintoimintojen  
suorittamiseksi. Käyttäjän ja langattoman viestintälaitteen 1 välinen  
kommunikointi voidaan suorittaa käyttöliittymällä 7, joka käsittää edulli-  
sesli näytön 8, näppäimistön 9 ja audiovälineet 10a, 10b, 10c. On sel-  
25 vää, että edellä esitetty langaton viestintälaite 1 on vain eräs pelkistetty  
esimerkki, mutta käytännössä langaton viestintälaite 1 voi käsittää  
 muitakin toiminnallisia lohkoja kuin edellä on esitetty. Myöskään digi-  
taalista signaalinkäsittely-yksikköä 3 ei välttämättä tarvita keksinnön  
mukaisessa langattomassa viestintälaitteessa, jolloin tarvittavat sig-  
30 naalinkäsittelytoiminnot on toteutettu muissa lohkoissa. Lisäksi langa-  
ton viestintälaite 1 käsittää kellopiirin 11 tai vastaavan mm. ohjausloh-  
kon toiminnassa tarvittavien kellosignaalien muodostamiseksi.

35

Kuvassa 2 on esitetty keksinnön erään edullisen suoritusmuodon mu-  
kaisen järjestelmän toiminnallisia lohkoja pelkistetysti. Tämä järjes-  
telmä muodostaa eräänlaisen ohjausarkkitehtuurin, jolla hallitaan is-  
tuntojen suoritukseen osallistuvien ohjelmaprosessien toimintaa ja lan-

- gattoman päätelaitteen resurssien käyttöä. Ohjelmaprosessilla tarkoitetaan sellaista ohjelmiston osaa, joka on käyttöjärjestelmän suorittaman ajoitustoiminnon kannalta itsenäisesti ajoitettavissa. Näitä resursseja ja resurssityyppejä on erilaisia eri käyttötarkoituksiin. Joitakin esimerkkejä näistä resursseista mainittakoon tässä lyhyesti: käyttöliittymän 7 eri toiminnot, kuten tietojen esittäminen näytöllä 8, tietojen lukeminen näppäimistöltä 9, audiosignaalien käsittelytoiminnot; tiedonsiirtotoiminnot, kuten erilaisten viestien vastaanottaminen ja muodostaminen sekä vastaussanomien muodostaminen; signaalinkäsittelytoiminnot; protokollaohjelmapinot; toiset sovellusistunnot. Järjestelmässä 10 on muistiin 4 muodostettu myös mm. ensimmäinen sanomapuskuri B1 kaikkia resurssivälittimistä RH(i), RH(j), RH(k), jne. istuntojen hallintaja ajoituslohkolle ASM (Application and Scheduling Manager) lähetettäviä sanomia varten sekä toinen sanomapuskuri B2 erityisesti 15 istuntojen aloituspyyntösanomien tallennusta varten. Järjestelmän muistiin 4 on vielä muodostettu mm. muistialue resurssienjakotaulun RAT (Resource Allocation Table) tallentamista varten. On selvää, että muistiin 4 voidaan käytännön sovelluksissa varata myös multa (dynaamisia ja/tai kiinteitä) muistialueita eri käyttötarkoituksiin.
- 20 Järjestelmän toiminta voidaan jakaa kolmeen perustoimintoon: istuntojen käynnistyksen hallinta (Access Control), istuntojen keskeytyksen hallinta (Suspension Control) sekä resurssien varaamisen hallinta (Resource Reservation Control). Käynnistyksen hallinnalla rajoitetaan 25 istuntojen suoritukseen sisältyvien ensimmäisten sovelluslohkosäiliöiden käynnistämistä huomioiden kulloinenkin kuormitustilanne sekä resurssien varaustilanne. Keskeytyksen hallinnalla ohjataan resurssien varaustilanteen mukaan käynnissä olevien istuntojen suorituksen lopusta tai lopullista keskeyttämistä yksittäisten sovelluslohkojen päätymiskohdissa. Resurssien varaamisen hallinnalla huolehditaan istuntoihin sisältyvien sovelluslohkojen tarvitsemien ja pyytämien resurssien sekä kulloinkin voimassaolevien resurssivarausten tietojen ylläpitämisestä ja välittämisestä istuntojen hallinta- ja ajoitustoiminnoille.
- 30
- 35 Edellä mainittujen perustoimintojen toteuttamiseksi on järjestelmään muodostettu joitakin toiminnallisia kokonaisuuksia. Keskeisenä järjestelmän toiminnallisena kokonaisuutena ovat sovellusistuntojen ja re-

## 10

- surssien hallintatoiminnot MG, jotka jakautuvat mm. sovellusistuntojen hallinta- ja ajoituslohkoon ASM ja resurssien hallinta- ja varauslohkoon RAM. Sovellusistuntojen hallinta- ja ajoituslohko ASM huolehtii mm. istuntojen käynnistämisen sallimisesta ja ajoituksesta, käynnistettyjen
- 5 istuntojen sisältämien tehtävien suorituksen käynnistämisen ajoituksesta sekä vastaavanlaisista sovellusriippumattomista toiminnoista. Tarkempi kuvaus tämän lohkon ja muidenkin järjestelmän lohkojen toiminnoista on esitetty jäljempänä tässä selityksessä. Resurssien hallintaa ja varauskirjanpitoa varten on järjestelmään muodostettu resurs-
- 10 sikohtaiset resurssien välittimet RH(i) (Resource Handler) sekä resurssien hallinta- ja varauslohko RAM (Resource Allocation Manager). Sovellusistuntojen sisältämät tehtävät on toteutettu sovelluslohkoina, jotka on sijoitettu järjestelmän suorituskvyn kannalta tarkoituksenmukaisella tavalla sovelluslokkosäiliöihin AC (Application block Container), joista
- 15 kukin voi sisältää yhteen tai useampaan istuntoon sisältyviä sovelluslokkoja AB (Application Block). Yksittäinen sovelluslokk AB voi eriaikaisesti suoritettuna sisältyä eri istuntoihin. Järjestelmää varten on vielä kehitetty istunnonohjausprotokolla (Session Control Protocol, SCP) ohjaamaan istunnon sovelluslokkien suorituksen käynnistämistä, kuten myöhemmin tässä selityksessä kuvataan. Vaikka kuvassa 2
- 20 on järjestelmää kuvattu erillisinä lokkoina, käytännössä kukin lokk on toteutettu edullisesti suorittimen 2 ohjelmakoodina, jotka sisältävät lokkien toiminnassa tarvittavat ohjelma-askleet. Järjestelmän suoritusta ohjataan käyttöjärjestelmän vuorottajan tai vastaavan avulla, joka
- 25 antaa järjestelmän ohjelmakoodit sisältäville ohjelmaprosesseille suoritusaikaa niille asetetun tärkeysjärjestyksen perusteella.

- Tässä selityksessä käytettävää nimitystä istunto (sovellusistunto) ei tule tulkita rajoitetusti, vaan istunto voi olla esim. yksittäiseen tehtävään
- 30 tarkoitettu yksinkertainen joukko ohjelmakäskyjä, joka muodostaa yhden sovelluslokkon, tai istunto voi koostua joukosta erillisiä ohjelmoduuleita tai vastaavia, jotka voidaan ryhmitellä yhdeksi tai useammaksi sovelluslokkoksi. Istunto voi myös olla riippuvainen muista istunnoista, jos esimerkiksi yksi sovellusistunto käynnistää istunnon jostakin
- 35 toisesta sovelluksesta saadakseen jälkimmäiseltä jonkin tarvitsemansa tiedon. Sovellusistunnosta (Application Session) voidaan käyttää myös nimitystä sovellus (Application). Langattomassa viestintälaitteessa suo-

## 11

- ritettavista istunnoista mainittakoon ei-rajoittavina esimerkkeinä kalenteri, puhelinmuistio, puheluun vastaaminen, tekstiviestien kirjoittaminen ja lähettäminen, tekstiviestien vastaanottaminen ja lukeminen, jne. Nyt esillä olevassa keksinnössä kukin istunto muodostetaan siten, että se käsittää yhden tai useamman tehtävän. Näitä tehtäviä vastaavien sovelluslohkojen suoritusjärjestys voi olla kiinteä, tai istunnon kuluessa sovellusistuntojen hallinta- ja ajoituslohko ASM voi määrätä, mikä sovelluslohko seuraavaksi suoritetaan. Valintaan vaikuttaa esimerkiksi edeltävän sovelluslohkon lopputulos tai tietyn tapahtuman esiintyminen, odottamattoman signaalin saapuminen, tms. Nämä samaan istuntoon sisältyvät sovelluslohkot voidaan sijoittaa esimerkiksi yhteen sovelluslokkosäiliöön, jos samanaikaisesti suoritettavia tehtäviä ei istuntoon sisälly.
- 5
- 10
- 15 Suorituksessa olevat istunnot muodostuvat joukosta peräkkäisiä ja/tai rinnakkaisia tehtäviä, joita vastaavat suoritettavat sovelluslohkot sijaitsevat yhdessä tai useammassa sovelluslokkosäiliössä riippuen rinnakkaisuusvaatimuksista. Esimerkiksi saapuvan puhelun tilanteessa käynnistetään peräkkäisesti joukko istuntoja, joilla voidaan ilmoittaa saapuvasta puhelusta, hakea soittavan numeron perusteella vastaavat nimetiedot langattoman päätelaitteen puhelinmuistiosta sekä ilmoittaa henkilön nimi langattoman viestintälaitteen näytöllä, ja odottaa, vastaako langattoman päätelaitteen käyttäjä saapuvaan puheluun, eli edullisesti jäädään odottamaan näppäinpainalluksia. Jos käyttäjä painaa esim.
- 20
- 25 luurinäppäintä, suoritetaan näppäinpainalluksen tulkinta ja sen perusteella päätetään jatkotoimista, kuten puheluun vastaamisesta. Edellä mainitun esimerkkitalanteen eri toiminnot voidaan toteuttaa erillisinä istuntoina, tai saman istunnon eri tehtävinä. Kukin mainittu tehtävä voidaan käsittää eräänlaiseksi istunnoksi. Yhden istunnon sisällä on eri tehtäviä vastaavien sovelluslohkojen tarvitsemien resurssien varaus ja vapautus kuitenkin joustavampaa kuin mitä se olisi tehtävien ollessa erillisiä istuntoja.
- 30
- 35 Seuraavassa selostetaan keskeisten toiminnallisten lohkojen pääasialliset tehtävät. Sovellusistuntojen hallinta- ja ajoituslohko ASM huolehtii samanaikaisesti suoritettavana olevien istuntojen sekä niihin sisältyvien sovelluslohkojen suorituksen lomittamisesta (ajoituksesta) toisiinsa

## 12

- nähdessä, siis kullekin sovelluslohkolle tarvittavan suoritusajan ja muiden resurssien varaamisen ajoitukseen tarvittavasta ohjauksesta. Sovellusistuntojen hallinta- ja ajoituslohko ASM huolehtii myös istuntojen istunto-  
5 istuntoprofiiliin (Application Session Profile) lukemisesta istunnon istunto-  
5 istuntoprofiilitaulusta APT (Application Profile Table). Istuntoprofiilitaulu APT sisältää tiedon, eräänlaisen tehtäväkaavion (Activity Diagram), istunnon peräkkäisten ja rinnakkaisten sovelluslohkojen (eli istunnon sisältämien tehtävien) keskinäisistä peräkkäisyys- ja rinnakkaisuusriippuvuuksista. Tämän istuntoprofiiliin perusteella sovellusistuntojen hallinta-  
10 ja ajoituslohko ASM tietää, miten istunnon suorituksen tulee edetä sovelluslohkosta toiseen. Samaa sovellusta varten on mahdollista laatia useampiakin kuin yksi sovellusistuntoprofiili esim. kahta erilaista käyttötilannetta varten, esim. työaikana ja vapaa-aikana. On huomattava, että sama sovelluslohko voi sisältyä useaan erilaiseen sovellusistuntoprofiiliin. Esimerkiksi voidaan käynnistää eri aikoina saman sovelluksen istuntoja useita, joissa kussakin istuntoprofiili on vain joidenkin tehtävien osalta erilainen. Saman sovelluslohkosäiliön sisältämiä lohkoja voidaan tällöin suorittaa eri istunnoissa tarvitsematta kopioida näitä lohkoja kutakin istuntoa varten. Tästä on etuna mm. se, että muistia 4  
20 tarvitaan vähemmän. Sovellusistunnoille määritetään edullisesti käynnistuksen yhteydessä mm. tärkeysjärjestys, jolla samanaikaisesti suoritettavana olevien istuntojen tehtävien suoritussyajrjestystä ja mahdollista keskeytystä voidaan ohjata. Lisäksi kullekin istunnon sovelluslohkolle on edullisesti määritetty arvioitu suoritus aika sekä kunkin sovelluslohkoon tarvitsemat resurssit. Näitä tietoja sovellusistuntojen hallinta- ja ajoituslohko ASM voi käyttää sovelluslohkoon ajoituksen ohjauksessa ja resurssitilanteen arvioimisessa erityisesti ylikuormitustilanteissa. Sovellusistuntojen hallinta- ja ajoituslohkolla ASM on käytettävissään sovelluslohkoon AB sijoittelutaulu BAT (Block Assignment Table),  
25 jossa on tieto siitä, mitä sovelluslohkota mikin sovelluslohkosäiliö sisältää ja mikä lohko on aktiivisesti suoritettavana missäkin istunnossa. Nämä tiedot on tallennettu sovelluslohkoon sijoittelutietueisiin BAR (Block Assignment Record).
- 30
- 35 Lisäksi sovellusistuntojen hallinta- ja ajoituslohkolla ASM voi olla käytettävissään tieto kunkin sovelluslohkoon kiinteästi tai mahdollisesti tarvitsemista resursseista sovellusistuntojen profiilitaulussa APT

- 5 (Application Profile Table), jossa on sovellusistunnon profiilitietoja APR (Application Profile Record). Jos kaikkien tarvittavien resurssien tietoja ei ole sovellusistuntoprofiilissa, niin ne käyvät selville sovelluslohkosten lähettäessä varauspyyntöjä resurssivälittimille ja resurssien hallinta- ja varauslohkolle.

- 10 Sovellusistuntojen hallinta- ja ajoituslohko ASM edullisesti tallentaa historiatietoa kunkin istunnon osalta. Tätä varten on järjestelmään edullisesti luotu istuntohistoriataulu SHT (Session History Table), jossa olevassa tietueessa SHR (Session History Record) ASM ylläpitää suorittavana olevan istunnon historiatietoa, joka on edullisesti käytettävissä kunkin sovelluslohkon tuottaman informaation siirtämiseksi seuraaville sovelluslohkoille.

- 15 Sovellusistuntojen hallinta- ja ajoituslohko ASM vastaanottaa uusien istuntopyyntöjen yhteydessä lähetettävät aloituspyyntösanomat, jotta sovellusistuntojen hallinta- ja ajoituslohko ASM voi tarkistaa kuormitus-tilanteen sekä huolehtia myös käynnistettävän istunnon sisältämien lehtävien käynnistykseen ajoituksesta. Erityisesti suorittimen tai muiden resurssien ylikuormitus-tilanteessa sovellusistuntojen hallinta- ja ajoituslohko ASM tekee päätöksen siitä, voidaanko aloitettavaksi pyydetty istunto käynnistää. Päätöksen tekemiseksi sovellusistuntojen hallinta- ja ajoituslohko ASM suorittaa sovelluslohkosten suoritus- ja ajoitus- ja resurssivarausten kesto- ja ajoitus- ja mahdollisesti muiden tietojen perusteella edullisesti sen tutkimisen, voidaanko aiemmin käynnistettyjen tärkeämpien istuntojen tosiaikaisuus taata myös uuden istunnon käynnistämisen jälkeen. Tällaisella järjestelyllä voidaan hallita ylikuormitus-tilanteet ja suuressa osin estää niiden syntyminen. Sovellusistuntojen hallinta- ja ajoituslohko ASM huolehtii myös istunnon tilapäisestä keskeyttämisestä jonkin suoritetun sovelluslohkon päättymiskohdassa osimorkiksi silloin, kun sama resurssi tarvitaan jonkin tehtävän suorittamiseen jossakin kilpailijassaan istunnossa tai kun jokin tarvittava resurssi on kokonaan varattuna parhaillaan suorituksessa olevien sovelluslohkosten loppuun saattamiseksi. Istunnon keskeyttäminen voidaan käytännössä toteuttaa siten, että edellisen sovelluslohkon päätyttyä sovellusistuntojen hallinta- ja ajoituslohko ASM ei lähetä ohjauskysymystä (Do sanoma) suoritusjärjestyksessä seuraavan

- samaan istuntoon kuuluvan sovelluslohkon sisältävälle lohkosäiliölle ennen kuin sille voidaan antaa suoritusaikaa. Joissakin tilanteissa voi istunto käsittää rinnakkain suoritettavaksi tarkoitettuja sovelluslohkota, jotka tällöin ovat kukin eri sovelluslohkosäiliössä. Mikäli mainittu istunnon keskeytyskohta on tällaisessa rinnakkaisen toiminnan toteuttavien sovelluslohkosten alussa, ei ohjauskäskysanomaa lähetetä millekään näistä sovelluslohkolta, ennen kuin istunnon toiminta voi jalkua.

- 10 Sovellusistuntojen hallinta- ja ajoituslohkon ASM tehtävänä on myös ohjata miltä tahansa resurssivälittimeltä tulevat sanomat oikeassa järjestyksessä oikeille sovelluslohkoille oikeissa sovelluslohkosäiliöissä silloin, kun mikään sovelluslohko ei ole synkronisesti odottamassa saapuvaa sanomaa. Tämä on tarpeen erityisesti synkronoimattomassa (asynkronisessa) sanomavälityksessä tilanteessa, jossa saman resurssin kapasiteettia on jaettu useammalle kuin yhdelle sovelluslohkolle tai kun vastaavan varauspyyntösanoman on lähettänyt sovelluslohko, joka ei itse jää odottamaan vastaavaa varausilmoitussanomaa (In sanoma). Muussa tapauksessa resurssivälittimeltä tuleva sanoma saattaisi ohjautua sellaiselle sovelluslohkosäiliölle tai sovelluslohkolle, jolle kyseinen sanoma ei ollut tarkoitettu. Sovellusistuntojen hallinta- ja ajoituslohko ASM osallistuu myös resurssien vapautukseen antamalla tarvittavan ohjauksen sovelluslohkoille mm. ylikuormitus- ja virhetilanteissa.
- 25 Resurssivälittimet pitävät yllä listaa kaikista tehdyistä resurssivarauksista (Resource reservation Instance, RI) vastaaviin resurssivaraustauluihin RIT tallennettavina resurssivaraustietueina (RIR, Resource Instance Record). Jonkin sovelluslohkosäiliön sisältämän sovelluslohkon pyytäessä resurssia käyttöönsä välitetään sovelluslohkosta suoraan asianomaiselle resurssivälittimelle RH tai resurssien hallinta- ja varauslohkolle RAM edullisesti tietoa siitä, minkälaisesta resurssitarpeesta on kyse. Esimerkiksi tiedonsiirtoa varten muodostettujen resurssien yhteydessä voi olla mahdollista valita tiedonsiirtonopeus, tiedonsiirrolta edellytettävä virhetodennäköisyys tms., joten resurssipyynnössä (varauspyyntösanomassa Con) voidaan välittää näihin resurssivarausten ominaisuuksiin vaikuttavia parametreja. Tällöin resurssivälitin RH tai resurssien hallinta- ja varauslohko RAM tutkii resurssipyynn-



## 15

nön vastaanotettuaan edullisesti sen, onko pyydettyjen parametrien mukaista resurssivarausta mahdollista tehdä resurssia pyytäneelle sovelluslohkolle. Aiemmin tehty resurssivaraus voi myös olla vapaana ja uudelleen varattavissa, jos se täyttää pyynnön sisältämät ominaisuus-

5 vaatimukset. Sovelluslohko tai sovelluslohosäiliö voi takaisinohjaus-

sanomalla vapauttaa hallussaan olleen resurssivarauksen, jolloin re-

surssivälitin RH merkitsee resurssivarauksen siinä mielessä vapaaksi,

että toinen sovelluslohko tai sovellusistunto voi pyytää resurssia varat-

10 tavaksi. Jos resurssin varaaminen ja/tai vapauttaminen vaatii joidenkin

toimenpiteiden suorittamista itse resurssilta, resurssivälitin RH on sel-

villä keskeneräisestä varaamisesta tai vapauttamisesta ja resurssia

pyytävä sovelluslohko tai sovellusistunto voi joutua odottamaan va-

rausilmoilussanomaa. Resurssien hallinta- ja varauslohkolla on tieto

kaikkien resurssityyppien resurssivaraustilanteesta. Tällä järjestelyllä

15 pyritään estämään syntymästä tilanteita, joissa jokin resurssi on yli-

kuormitettu.

Sovellusistuntojen hallinta- ja ajoitusvälineet (ASM) ja resurssien hal-

linta- ja varausvälineet on sopivimmin muodostettu välitilattomiksi, jol-

20 loin istuntojen tietoja koskevat muutokset tallennetaan mainittuun istun-

tohistoriatauluun (SIIT), ja resurssivarausten tietoja koskevat muutok-

set tallennetaan mainittuun resurssienjakotauluun (RAT). Myös resurs-

sivälittimet on sopivimmin muodostettu välitilattomiksi, jolloin ne tallen-

tavat resurssivarauksen tietoja koskevat muutokset resurssivaraustau-

25 luun (RIT)

Samaan resurssiin voi olla tehtynä useampiakin kuin yksi resurssiva-

raus esim. tilanteessa, jossa useammalla sovelluslohkolla yhdessä tai

useammassa istunnossa on tarve käyttää samaa resurssia olennaisesti

30 samanaikaisesti. Nämä samaa resurssia käyttävät sovelluslohkot eivät

yleensä ole samassa lohosäiliössä, vaan ne voivat olla eri säiliöissä ja

eri istunnoissa. Tällaisessa tilanteessa on tärkeää huolehtia siitä, että

resurssivälittimelle menevät varauspyyntösanomat (Con sanomat) ja

resurssivälittimellä tulevat varausilmoilussanomaiset (In sanomat) kye-

35 tään liittämään toisiinsa. Tätä varten on resurssien hallinta- ja varaus-

lohkoon RAM muodostettu toiminnot, joilla RAM suorittaa samaan re-

surssivälittimeen kohdistuvien resurssivarauspyyntöjen käsittelyn, ku-

## 16

ten jonotuksesta huolehtimisen ja pyyntöjen välittämisen oikea-aikaisesti ja oikeassa järjestyksessä resurssivälittimelle. Resurssivälittimeltä tulevien varausilmoitussanomien välittäminen oikealle lohkosäiliölle ja oikeaa varauspyyntösanomaa vastaavasti perustuu siihen, että pyynnön lähettänyt sovelluslohkosäiliö aina aktiivisena odottaa vastaavaa varausilmoitussanomaa ja kukin varauspyyntösanoma sisältää yksikäsittelyn viitetunnisteen. Jokaisen varauspyynnön lähetyksen yhteydessä sovelluslohko varustaa pyynnön viitetunnisteella ja informoi ja välittää pyynnöstä ja sen viitelunnisteesta tiedon sovellusistuntojen hallinta- ja ajoituslohkolle ASM tallennettavaksi viitetunnisteen korrelaatiotauluun CRT (Correlation Reference Table) viitetunnisteen korrelaatiotietueeksi CRR (Correlation Reference Record). Viitetunnisteen käytöllä mahdollistaa usean varauspyynnön olemassaolon samanaikaisesti ja varausilmoitussanomien odottamisen myös asynkronisesti.

15

Samaan viitetunnisteen korrelaatiotauluun CRT tallennetaan myös sellaisten sovellussanomien viitetunnisteen, joihin sovellusistunto odottaa vastaussanomaa jonkin resurssivälittimen (esim. tietoliikenneyhteyden) kautta. Viitetunnisteen käyttö mahdollistaa usean pyynnön olemassaolon samanaikaisesti ja odottamisen asynkronisesti.

20

Sovelluslohko itse huolehtii myös resurssivarausten vapauttamisesta silloin, kun sovellusistuntojen hallinta- ja ajoituslohko ei ole määrännyt resurssivarausta pidettäväksi varattuna saman istunnon seuraavaa tai myöhempiä sovellusloikkoja varten. Mahdollisissa vikatilanteissa sovellusistuntojen hallinta- ja ajoituslohko ASM käynnistää suoritettavaksi tarvittavan sovellusriippumattoman vapautustoiminnon. Tällöin resurssi ei jää tarpeettomasti varatuksi.

25

Varsinainen resurssivarausten pyyntösanomien käsittely suoritetaan resurssivälittimillä, joita on muodostettu edullisesti yksi kutakin resurssityyppiä varten. Resurssivälitin RH(i) on ohjelmapirosessi (käsittelijä, handler), joka on tarkoitettu tiettyä resurssia koskevien varaus-, vapautus- ja muiden ohjauspyyntöjen käyttämiseksi samalla tavalla kaikkien erilaisten istuntojen ja erilaisten resurssityyppien tapauksessa. Kutakin resurssityyppiä i varten muodostetussa resurssivälittimessä RH(i) on huomioitu kyseisen tyyppin erityispiirteet, esim. videon käsit-

35

- tely, audiosignaalien käsittely, tekstiviestien käsittely, jne. Resurssivälittin ylläpitää listaa kyseisen resurssin käyttämiseksi muodostetuista resurssivarauksista. Resurssivälittimien suoritus on ajoitettu sovellusistuntojen puolelta määräytyvän tarpeen mukaan käyttäen hyväksi käyttöjärjestelmän sisältämää vuorottajaa.

- Sovelluslohkosäiliö AC (application block container) on yhteen tai useampaan erilaisoon sovellusistuntoprofiiliin sisältyvien sovelluslohkojen sekä sovelluslohkosäiliön kuorirakenteen muodostama ohjelmaprosessi. Sovelluslohkoja on yleensä useita ja niiden suoritus on saman säiliön sisällä peräkkäistä, mutta eri säiliöiden tapauksessa rinnakkaisista. Kussakin sovelluslohkosäiliössä on yhtenä lohkona odotuslohko IB (Idling Block) asynkronisten lapahtumien odottamista varten. Lisäksi kussakin lohkossa on sisäänmenomoduuli ESM (Entrance State Module), haarautumismoduuli BSM (Branching State Module), sekä ulosmenomoduuli TSM (Termination State Module) lohkosäiliön suorituksen lopettamista varten (eli siirtämiseksi pois aktiivisesta tilasta). Lisäksi kunkin sovelluslohkon alussa on aloitustilamoduuli SSM (Start State Module) ja lopussa pysäytysmoduuli SM (Stop Module). Sovelluslohkosäiliö AC sisältää myös mm. listan lohkosäiliön toiminnassa tarvittavista muuttujista, tietorakenteista, muistivarauksista, tietoa resurssivarauksista yms. Lohkosäiliö AC vastaanottaa hallinta- ja ajoituslohkolta ASM lähetettäviä sanomia sovelluslohkojen suorituksen käynnistämiseksi. Lisäksi lohkosäiliön sisältämät edellämainitut moduulit ja odotuslohko välittävät istunnonohjausprotokollaan kuuluvia takaisinohjaussanomiamia (Out sanoma) resurssien hallinta- ja varauslohkolle RAM, joka välittää ne edelleen sovellusistuntojen hallinta- ja ajoituslohkolle ASM. Nämä takaisinohjaussanomat liittyvät esimerkiksi tilanteisiin, joissa sovelluslohkon suoritus on siirtynyt odottamaan jonkin ehdon toteutumista, esim. sovellussanomien saapumista langallomaan viestintälaitteeseen, tai lohkon suoritus on päättynyt.

- Nyt esillä olevan keksinnön mukaiseen järjestelmään kehitettyä istunnonohjausprotokollaa SCP käytetään mm. järjestelmän toiminnallisten lohkojen yhteistoiminnan aikaansaamiseen, perustuen istunnonohjaussanomien välitykseen toiminnallisten lohkojen välillä. Kyseessä on eräänlainen viestien välitystoiminto sovellusistunnon suoritukseen

## 18

osallistuvien toiminnallisten lohkojen välillä. Istunnonohjausprotokollalla SCP voidaan hallita langattoman viestintälaitteen suorittimen kuormitusta ylikuormitustilanteiden välttämiseksi ja hallitsemiseksi. Lisäksi istunnonohjausprotokollalla voidaan hallita yksittäisten toimintojen käynnistuksen ajoitusta sovellusistunnon aikana, ja välittää tarvittavia istuntoon osallistuvien toiminnallisiin lohkoihin kuuluvien ohjelmaprosessien tunnuksia (PID, Process Identifier) sekä muuta sovellusriippumatonta tietoa toiminnallisten lohkojen välillä. Istunnonohjausprotokollalla voidaan järjestää sovellusistuntojen keskinäiset priorisoinnit esimerkiksi istuntojen sisältämien tehtävien ajoituksen perusteella. Jaettujen resurssien, kuten tietoliikenneyhteyksiin liittyvien resurssien, varausten järjestäminen optimaalisella tavalla on keksinnön mukaisessa järjestelmässä tehty mahdolliseksi sovellusistuntojen hallinta- ja ajoituslohkolle istunnonohjausprotokollan avulla.

15

Istunnonohjausprotokolla on pyritty toteuttamaan siten, että resurssien varaamiseen liittyvät toiminnot, kuten resurssivarauspyyntösanomat (Con sanomat), voidaan toteuttaa erilaisissa sovellusistunnoissa mahdollisimman samanlaisella tavalla resurssin tyypistä riippumatta. Tämä helpottaa mm. sovelluslohkojen ohjelmakoodien laatimista.

20

Istunnonohjausprotokollassa käytetään erilaisia sanomia järjestelmän ohjaamiseksi ja tietojen välittämiseksi eri toiminnallisten lohkojen välillä. Näitä sanomia nimitetään tässä yhteydessä yhteisellä nimityksellä istunnon ohjaussanomat, ja niitä ovat mm. ohjauskäskyosanoma Do, takaisinohjaussanoma Out, varauspyyntöosanoma Con ja varauslmoitussanoma In. Lisäksi poikkeustilanteita varten tarvitaan omat sanomansa, jolta ei kuitenkaan ole tarve käsitellä enempää tässä yhteydessä. Tietorakenne voi eri sanomilla olla erilainen, mutta kukin sanoma sisältää edullisesti tiedon siitä, mihin sovellusistuntoon sanoma liittyy sekä miltä ohjelmaprosessilta se on lähetetty ja/tai mille ohjelmaprosessille se on tarkoitettu. Sanomissa on tyypillisesti myös yksi tai useampi tietokenttä, jossa välitetään sanomatyypin mukaista tietoa sanoman vastaanottajalle. Istunnonohjausprotokollan sanomissa välitetään mm. dynaamisesti syntyvien tai valikoitujen samaan istuntoon osallistuvien resurssivälittimien ja sovelluslohkosäiliöiden (ohjelmaprosessien) tunnistetta.

25

30

35

- Kuten edellä jo on todettu, sovelluslohkon käynnistys tapahtuu sovellusistuntojen hallinta- ja ajoituslohkon ASM lähettämällä ohjauskäskysanomalla (Do sanoma). Ohjauskäskysanomassa voidaan ilmoittaa
- 5 sovelluslohkolle, mitä resursseja se voi varata. Lisäksi sovelluslohkolle voidaan ohjauskäskysanomassa ilmoittaa, mitä resursseja sovelluslohkoon on varattava tai vapautettava. Esimerkkinä mainittakoon tilanne, jossa ollaan muodostamassa puhelua, ja tarvittavat tiedonsiirtoresurssit ovat vapaana. Tällöin puhelun muodostamisessa käytettävälle sovel-
- 10 luslohkolle voidaan ilmoittaa, että sen on varattava puhelussa tarvittavat tiedonsiirtoresurssit. Sen jälkeen kun sovelluslohko on suoritettu loppuun, lähettää se takaisinohjaussanomana (Out sanoma) resurssien hallinta- ja varauslohkoon RAM. Tässä takaisinohjaussanomassa on tarvittaessa mm. tietoa siitä, mitä resursseja sovelluslohko on varannut
- 15 ja/tai vapauttanut. Lisäksi takaisinohjaussanoma (Out sanoma) lähetetään niille resurssivälittimille, joiden resurssivaroja sovelluslohko vapauttaa joko toisille saman istunnon sovelluslohkoille tai toisille sovellusistunnoille. Saman istunnon myöhempi sovelluslohko voi varata olemassa olevan vapaan resurssivaroja käyttöönsä lähettämällä
- 20 resurssivälittimelle takaisinohjaussanomana (Out sanoma), jossa resurssivälitin saa tämän uudelleenvarauksen merkiksi asianomaisen sovelluslohkosäiliön tunnisteen. Kullakin resurssivälittimellä on kerrallaan tallessa ainoastaan yhden lohkosäiliön tunniste tai tämän puuttuessa sovellusistuntojen hallinta- ja ajoituslohkon tunniste tai muun resurssityyppiä vastaavan oletuslohkon tunniste. Edellä mainittu järjes-
- 25 tely mahdollistaa sen, että kullakin yksittäisellä resurssivaroja on erillinen näkyvyys (visibility) sovelluslohkosäiliöihin. Koska resurssien hallinta- ja varauslohkoon RAM suorituksen jälkeen suoritusvuoroon tulee olennaisesti välittömästi sovellusistuntojen hallinta- ja ajoituslohko
- 30 ASM korkean prioriteetin ansiosta, on resurssien käyttö- sekä varaus-tilanne tällä hetkellä ainakin osittain ajantasalla sovellusistuntojen hallinta- ja ajoituslohkoon ASM varien. Tällöin sovellusistuntojen hallinta- ja ajoituslohko ASM voi selvittää sen, mitä sovelluslohkota voidaan käynnistää ja mitä resursseja niille voidaan antaa käyttöön.
- 35
- Yhteenvedona resurssien varaamisesta todettakoon tässä yhteydessä se, että sovelluslohko voi saada käyttöönsä välittömästi tarvitsemansa

## 20

resurssin sovelluslohkon suorituksen aikana, mikäli resurssi on vapaana. Tällöin sovelluslohko lähettää varauspyyntösanoman pyydettävän resurssin tyyppiä vastaavalle resurssivälittimelle, joka tutkii, voidaananko pyydetty resurssi varata sitä pyytäneelle sovelluslohkolle.

- 5 Vastauksena resurssivälitin lähettää sovelluslohkolle varausilmoitus-sanoman, jossa on joko myönteinen kuittaus (ACK), jos resurssi voidaan antaa kyseisen sovelluslohkon käyttöön, tai kielteinen kuittaus (NACK), jos resurssia ei sillä hetkellä voida antaa kyseisen sovelluslohkon käyttöön. Olettaen, että sovellusistuntojen hallinta- ja ajoituslohkolta
- 10 ASM sovelluslohkolle tullessa ohjauskäskysanomassa annettu resurssien varausta ohjaava informaatio on täsmällistä, ei resurssivälittimiltä ylönsä tule kielteistä varausilmoitus-sanomaa. Mikäli resurssitarve ei välttämättä ole välitön tai edellä mainitussa ohjauskäskysanomassa on ilmaistu resurssin olevan esim. muussa käytössä varattuna,
- 15 sovelluslohko voi lähettää varauspyyntösanoman resurssien hallinta- ja varauslohkolle RAM, joka soveltaa jonotusperiaatetta. Tässä vaiheessa se sovelluslohosäiliö, jonka sovelluslohko lähetti varauspyyntösanoman resurssien hallinta- ja varauslohkolle RAM, jää odottamaan varausilmoitus-sanomaa joko synkronisesti tai asynkronisesti. Myöhemmin
- 20 resurssien hallinta- ja varauslohko RAM lähettää resurssien varaustilanteen salliessa odottamassa olleen varauspyyntösanoman asianomaiselle resurssivälittimelle, joka toimittaa varausilmoitus-sanoman suoraan resurssia pyytäneelle sovelluslohosäiliölle. Tämän mahdollistamiseksi sovellusistuntojen hallinta- ja ajoituslohko ASM voi joutua
- 25 lähettämään esim. rinnakkaisten sovellusistuntojen sovelluslohosäiliöille ohjauskäskysanomaa (Do sanomia) joidenkin resurssivarausten vapauttamiseksi.

- 30 Nyt esillä olevan keksinnön mukaiset toiminnot ovat suurelta osin toteutettavissa ohjelmallisesti langattoman viestintälaitteen 1 suorittimen 2 ohjelmakoodina. Resurssien hallinnan ja varauksen sekä resurssien välittimien voidaan ajatella muodostavan eräänlaisen välkeroksen (middleware) langattoman viestintälaitteen 1 käyttöjärjestelmän ja sovellusten välillä. Käyttöjärjestelmän kannalta keksinnön mukaisen järjestelmän toiminnot ovat yhden tai useamman ajastettavan säikeen (thread) suorittamista. Keksintöä ei ole sidottu mihinkään käyttöjärjestelmään, vaan keksintöä voidaan soveltaa sellaisten käyttöjärjestelmien
- 35

## 21

yhteydessä, joissa on mahdollisuus moniajioon. Käyttöjärjestelmä on kuitenkin edullisesti sellainen, jossa moniajtoa suoritettaessa yhden ajastettavissa olevan ohjelman prosessin (sälkeen) suorituksen keskeyttäminen ja toisen ohjelman prosessin suorituksen jatkaminen ei voi tapahtua missä tahansa kohdassa ohjelmakoodia, vaan ainoastaan ennalta määrätyissä paikoissa. Ohjelman laatija valitsi ja asettaa kohdat, joissa käyttöjärjestelmä saa vaihtaa suorituksessa olevan ohjelman prosessin. Tällaiset keskeytyspaikat on edullista sijoittaa sellaisiin kohtiin, joissa sovelluslohkosäiliön tai koko sovellusistunnon suoritus saattaa luonnollisesta syystä keskeytyä, esimerkiksi odotettaessa vastausta joltakin resurssivälittimestä.

Selostetaan seuraavaksi keksinnön erään edullisen suoritusmuodon mukaisen menetelmän toimintaa viitaten samalla kuvan 4 esimerkitilanteeseen. Kuvassa yhtenäisillä viivoilla esitetyt nuolet kuvaavat sanomien kulkua lohkojen välillä, katkoviivoitetut nuolet kuvaavat sovellusistunnon suorituksen etenemistä lehtävä lehtävältä, ja pisteviivoitetut nuolet kuvaavat sovellusistunnon ohjaussanomien kulkua lohkojen välillä.

Oletetaan, että langaton viestintälaitte 1 on käynnissä ja että käyttöjärjestelmä on käynnistynyt. Käyttöjärjestelmä käynnistää jossakin vaiheessa keksinnön mukaisen järjestelmän toiminnan edullisesti käynnistämällä sovellusistuntojen hallinta- ja ajoituslohkon ASM, resurssien hallinta- ja varauslohkon RAM sekä istunnonohjausprotokollan SCP alustustoiminnot. Siinä vaiheessa kun jokin sovellusistunto on tarkoitus käynnistää, välitetään sovellusistunnon aloituspyyntö hallinta- ja ajoituslohkolle ASM. Sovellusistunnon käynnistyminen voi johtua jonkin toisen sovellusistunnon lähettämästä aloituspyynnöstä, ulkoisesta tapahtumasta, kuten langattomaan päätelaitteeseen 1 saapuvasta kutsu-yrityksestä, langattoman päätelaitteen 1 käyttäjän suorittaman valinnan perusteella, tms. Kuvan 4 tilanteessa on langattomaan päätelaitteeseen 1 saapunut tekstiviesti, joka on aiheuttanut uuden sovellusistunnon käynnistämisen. Tekstiviestin vastaanottavaa yhteystyyppiä varten tarkoitettu resurssivälitin RH(1) on välittänyt sovellusistunnon aloituspyyntösanoman S1, joka on sijoitettu ensimmäiseen sanomapuskuriin B1. Hallinta- ja ajoituslohko ASM havaitsee sanomapuskuriin B1 saa-

## 22

puneen sanoman, lukee sanomasta pyydetyn sovellusistunnon identifioimiseen tarvittavan tiedon, sekä lukee käynnistettävän sovellusistunnon istuntoprofiiliin istuntoprofiilitaulusta APT. Myös aloitettavan sovellusistunnon historiatiedot alustetaan tässä yhteydessä istuntohistoria-

5 taulussa SHT. Sovellusistunnon Istuntoprofiilista (Application Session Profile, ASP) käyvät ilmi mm. istuntoon sisältyvät tehtävät sekä mahdollisesti niiden tarvitsemat resurssivaraukset, sekä tehtäviä vastaaville sovelluslohkoille mahdollisesti määritetyt arvioidut suoritusajat. Hallinta- ja ajoituslohko ASM suorittaa käynnissä ja keskeytellynä olevien

10 istuntojen istuntoprofiilien, resurssienjakotaulun RAT (Resource Allocation Table) tietojen sekä käynnistettävän istunnon istuntoprofiiliin perustolla analysoinnin sen selvittämiseksi, onko kyseinen istunto mahdollista käynnistää. Hallinta- ja ajoituslohko ASM arvioi suorittimen 2 kuormituksen lisäyksen, jos istunto käynnistetään. Lisäksi hallinta- ja

15 ajoituslohko ASM tutkii käynnistettävän istunnon tarvitsemien resurssien varaustilanteen ja arvioi myös näiden resurssien kuormitustilanteen. Mikäli hallinta- ja ajoituslohko ASM päättää, että istunto voidaan käynnistää, valitsee ASM jonkin kyseisen istunnon istuntoprofiiliin ensimmäisen sovelluslohkon sisältävän sovelluslohosäiliön AC. Valittu

20 sovelluslohosäiliö AC ladataan muistiin 4 ensimmäisen sovelluslohkon suoritusta varten, ellei se ole ennestään valmiiksi muistissa ja vapaana suoritukseen. Kuvan 4 mukaisessa esimerkissä käynnistettävä sovellusistunto sisältää joukon tehtäviä, joita vastaavat sovelluslohkot ovat sovelluslohkot 0B—11B. Istunnon suoritus aloitetaan käynnistämällä

25 ensimmäinen sovelluslohko 0B siinä vaiheessa, kun tämän sovelluslohon sisältävä edellä ladattu sovelluslohosäiliö AC tulee suoritustuoroon. Kuten jo aikaisemmin tässä selityksessä on todettu, voidaan sovellusistunnoille määrittää tärkeysjärjestys, jonka perusteella istuntojen saamaa suoritusaikaa sekä istuntojen suorituserjestyistä voidaan

30 säätää.

Sovellusistuntojen hallinta- ja ajoituslohko ASM siirtää sanomapuskurista B1 sinne saapuneen sovellusistunnon aloituspyyntösanoman S1 toiseen sanomapuskuriin B2 olennaisesti heti saatuaan suoritusaikaa

35 käyttöjärjestelmästä. Siinä vaiheessa kun uuden sovellusistunnon suoritus voidaan aloittaa, lähettää hallinta- ja ajoituslohko ASM ohjauskysymys (Do sanoma) valitulle kyseisen istunnon ensimmäisen so-



## 23

- velluslohkon 0B sisältävälle lohkosäiliölle AC. Tämä alkaansaa sovel-
- luslohkon 0B suorituksen käynnistämisen. Tässä esimerkissä sovel-
- luslohkon 0B suorituksen tultua valmiiksi ja vastaavan takaisinohjaus-
- sanoman (Out sanoma) tultua lähetetyksi resurssien hallinta- ja va-
- 5 rauslohkolle RAM, sovellusistuntojen hallinta- ja ajoituslohko ASM  
(saatuana tämän takaisinohjaussanoman resurssien hallinta- ja va-  
rauslohkolta RAM) lähettää sovellusistuntoprofiilin perusteella ohjaus-  
käskysanoman kolmelle sovelluslohkosäiliölle AC1, AC2, AC3 vastaa-  
via kolmea sovelluslohkoa 1B, 2B, 3B varten, jotka on tarkoitettu eri
- 10 tyyppisten yhteyksien muodostamiseksi rinnakkaisesti. Tässä esimer-  
kissä nämä yhteydet ovat vastaavasti audio-, data- ja video-tyyppisiä.  
Kukin sovelluslohko 1B, 2B, 3B lähettää varauspyyntösanoman (Con  
sanoma) omaa yhteystyyppiään vastaavalle resurssivälittäjälle RH(2),  
RH(3), RH(4). Näissä varauspyyntösanomissa on tieto, kuten ohjelma-
- 15 prosessitunnus tai muu yksilöivä tunnus, sanoman lähettäneestä loh-  
kosäiliöstä. Tämän tiedon avulla resurssivälitin voi lähettää varausil-  
moitussanoman (In sanoma) takaisin oikealle lohkosäiliölle. Resurssi-  
välitin lähettää takaisinpäin varausilmoitussanoman (In sanoma) siinä  
vaiheessa, kun resurssivälitin on välittänyt lohkosäiliöltä tulevan va-
- 20 rauspyyntösanoman (Con sanoma) yhteyden muodostavalle protokol-  
laohjelmalle sekä saanut läältä takaisin kuittauksen yhteyden valmistu-  
misesta. Tällöin lohkosäiliössä oleva sovelluslohko saa joko kielteisen  
tai myönteisen varausilmoituksen, eli tiedon siitä, että resurssivälitin on  
kapasiteettirajoitusten puitteissa myöntänyt resurssivarauksen sovel-
- 25 lusistunnolle ja että resurssivälitin on suorittanut mahdolliset muut re-  
surssin käyttämisessä tarpeelliset alustus- ja kirjaamistoimenpiteet. Li-  
säksi resurssivälittimen tehtävänä on välittää vastaavalta protokolla-  
ohjelmalta vastaanottamiaan sovellussanomiam sille sovellusistunnon  
suoritukseen osallistuvalla lohkosäiliölle, jonka tunnistetiedon resurssivälitti-
- 30 mellä kulloinkin on hallussaan. Sovellusistunto edullisesti siirtyy odo-  
tustilaan siinä vaiheessa, kun se resurssipyyntösanoman (Con sa-  
noma) lähettämisen jälkeen jää odottamaan resurssivälittimeltä tulevaa  
varausilmoitussanomaa (In sanoma). Tieto tästä odotustilaan siirtymi-  
sestä välitetään resurssien hallinta- ja varauslohkolle RAM edullisesti
- 35 takaisinohjaussanoman (Out sanoma) avulla kunkin varauspyyntösa-  
nomien lähelyksen yhteydessä. Kuvassa 4 sovelluslohkossa 4B odo-  
tetaan, että saadaan varausilmoitussanoma vastaukseksi kaikkiin kol-

## 24

meen varauspyyntösanomaan, ennen kuin siirrytään langattomasta päätelaitteesta 1 eri yhteystyyppettä myöten lähetettävien tietojen muodostukseen. Yleisen sovellusistunnon tapauksessa eri yhteystyyppettä käyttäen voi sovellusistunto suorittaa interaktiivista sovellussanomien vaihtoa sovelluslohkojen 5B, 6B ja 7B toimesta yhteyksien toisissa päissä olevien sovellusten kanssa, jolloin on edullista, että sovelluslohkot 5B, 6B ja 7B sijaitsevat kolmessa eri lohkosäiliössä. Yleisessä tapauksessa sovellusistunto voi sisältää lukumääräisesti paljon useampia rinnakkain suoritettavia sovelluslokoja.

10

Resurssivälittimen suorituksen ajoitus on järjestetty käyttöjärjestelmän toimintojen ja ajoituksen avulla siten, että resurssivälittimien toimintoja suoritetaan sen jälkeen, kun sovelluslohot ovat lähettäneet niille sanomia käsiteltäväksi. Resurssivälittimille on muodostettu rajapinta, jonka kautta sovellussäiliöt voivat välittää istunnonohjaussanomiam ja sovellussanomiam resurssivälittimille ja vastaavasti vastaanottaa sanomia resurssivälittimiltä. Tämä rajapinta on sovellusriippumaton ja olennaisesti myös resurssityyppiinriippumaton, jolloin sovellusistunnon ohjausprotokollan sanomiam voidaan käyttää olennaisesti vakioimuotoisiam sanomia. Tämän lisäksi resurssivälitin välittää sekä sovellussäiliöiltä resurssille päin että resurssilla sovellussäiliöille ja/tai sovellusistuntojen hallinta- ja ajoituslokolle ASM päin sovellussanomiam. Kutakin resurssivarauspyyntöä varten resurssivälittimet muodostavat resurssivarauksen (Resource Reservation Instance, RI), joka on eräänlainen määrittely kyseisestä resurssin varaukerrasta sekä resurssin käyttämiseen liittyvistä parametreista, kuten tiedonsiirtonopeus, näytön resoluutio, stereo/monoääni, jne. Resurssivarauskiam tallennetaan resurssivaraustauluun RIT (Resource Instance Table) resurssivaraustietueiksi RIR (Resource Instance Record). Kullakin resurssivälittimellä on oma resurssivaraustaulunsa. Kukin resurssivälitin huolehtii omista resurssivarauksistaan ja välittää niistä tiedon resurssien hallinta- ja varauslokolle RAM. Resurssien hallinta- ja varauslokokssa RAM on siis kaikkien käytössä olevien resurssivarausten tiedot edullisesti resurssienjakotaulussa (Resource Allocation Table, RAT).

35

Resurssien hallinta- ja varausloko RAM välittää resurssienjakotaulun sisältämät tiedot istuntojen hallinta- ja ajoituslokolle ASM, joka käyttää

## 25

näitä tietoja mm. eri sovellusistuntojen suorituksen saattamiseksi esim. keskeytystilasta suoritustilaan. Resurssien hallinta- ja varauslohko RAM vastaanottaa myös sovellusloHKosäiliöiltä tulevat takaisinohjaussanomat, kuten tiedon sovellusloHKon suorituksen päättymisestä, tai

5 sovellusloHKon sisällä alkavasta synkronisesta odotustilasta, tai varausilmoitussanoman tulemisesta resurssivälittimeltä jollekin sovellusloHKosäiliölle.

Sen jälkeen kun resurssien hallinta- ja varausloHKo RAM on välittänyt resurssienjakotaulun tiedot sovellusistuntojen hallinta- ja ajoitusloHKolle

10 ASM, antaa suoritin 2 suoritusvuoron sovellusistuntojen hallinta- ja ajoitusloHKolle ASM, joka valitsee seuraavaksi suoritusvuoroon tulevan sovellusistunnon ja sovellusloHKon. Valinnassa sovellusistuntojen hallinta- ja ajoitusloHKo ASM käyttää perusteena mm. resurssien varausilannella sekä suorittimen 2 kuormitusilannetta. Valitulle tällaisen so-

15 vellusloHKon sisältävälle loHKosäiliölle lähetetään edullisesti ohjauskäs-kysanoma (Do sanoma), jolloin sovellusloHKon suoritus käynnistetään. Samanaikaisesti voi useampikin kuin yksi eri loHKosäiliöissä sijaitseva sovellusloHKo olla suoritusvuorossa (ns. rinnakkaiset ohjelmaproses-

20 sit). Käytännössä tämä merkitsee sitä, että suoritin 2 vuorottelee näille sovellusloHKosäiliöille annettavaa suoritusaikaa. Esimerkkinä tällai-sesta rinnakkaisuudesta ovat kuvan 4 esimerkissä esitetyt sovellusloH-

25 kot 1B, 2B, 3B, sekä myös 5B, 6B, 7B ja 8B, 9B, 10B. Moniprosessori-järjestelmissä on kuitenkin mahdollista myös todellinen samanaikai-suus sovellusloHKosäiliöiden suorittamiseksi.

Oletetaan, että suoritusvuorossa ovat osaohjelmat 5B, 6B, 7B, jotka muodostavat langattomasta päätelaitteesta 1 lähetettävän audio-, data- ja videotyyppisen tietosisällön sekä tiedonsiirron ohjaukseen tarvittavat

30 sovellussanomat. Nämä sovellusloHKot 5B, 6B, 7B suorittavat siis siir-rottävän tietosisällön muodostuksen ja tarvittavien sovellussanomien välittämisen vastaaville resurssivälittimille RH(2), RH(3), RH(4). Näitä sanomia on merkitty viitoilla D, E, F kuvassa 4. Resurssivälittimet suo-rittavat tarvittavat toimenpiteet, joiden tuloksena kolmea eri tyyppiä ole-

35 vat tietosisällöt lähetetään langattomasta päätelaitteesta 1. Sen jälkeen kun nämä kolme tiedonsiirtoa on tehty, muodostetaan sovellusloHKois-sa 8B, 9B, 10B takaisinohjaussanomat (Out sanomat), joilla vapaute-

## 26

- taan vastaavat kolme resurssivarausta. Sovellusistunnon suoritus voidaan tämän jälkeen päättää suorittamalla lopetuslohko 11B. Vapautuneille lohkosäiliöille AC1, AC2, AC3 varatut muistialueet voidaan tyhjentää ja päättyneen istunnon käyttämien resurssivarausten tiedot poistaa resurssivaraustauluista RIT ja resurssienjakotaulusta RAT.

Seuraavaksi tulevan sovellusistunnon aloituspyyntösanoman S2 käsittely suoritetaan edellä esitettyjä periaatteita noudattaen.

- 10 Kuvassa 3 on esitetty vielä erään sovelluslohkosäiliön AC rakennetta ja sovelluslohkosten suorituksen etenemistä sovelluslohkosäiliössä AC. Sovelluslohkosäiliössä on sisäänmenomoduuli ESM (Entrance State Module), joka aina suoritetaan ensimmäisen ohjauskäskysanoman tullessa vapaana olleelle lohkosäiliölle. Tämän jälkeen sovelluslohkosäiliössä on haarautumismoduuli BSM (Branching State Module), jossa tarkastetaan seuraavaksi suoritettavan sovelluslohkoston tunnisteen selvittämiseksi, minkä sovelluslohkoston AB1, AB2, AB3 alkuun lohkosäiliön suorituslogiikan tulee haarautua. Kunkin sovelluslohkoston lopussa on pysäytysmoduuli SM (Stop Module), joka ilmaisee lohkosten suorituksen päättymisen resurssien hallinta- ja varauslohkoston RAM takaisinohjaussanomalla (Out sanoma) ja sitä kautta edelleen istuntojen hallinta- ja ajoituslohkoston ASM, minkä jälkeen lohkosäiliön AC suorituslogiikka siirtyy odottamaan seuraavaa ohjauskäskysanomaa (Do sanoma) voidakseen jatkaa istunnon suoritusta järjestyksessä seuraavasta sovelluslohkoston siinä vaiheessa kun käyttöjärjestelmä antaa tälle lohkosäiliölle suoritusaikaa. Tätä varten suorituslogiikka siirtyy pysäytysmoduulista SM odotuslohkoston IB alkuun, missä on odotustilamoduuli ISM (Idling State Module), joka on tarkoitettu ottamaan vastaan kaikkia mahdollisia resurssivälittimiltä sovelluslohkosäiliölle tulevia sanomia.
- 25 Tällaisen sanoman vastaanotto sanomapuskuriin tapahtuu odotuslohkoston toimesta, minkä jälkeen odotuslohkoston lopussa oleva päätösmoduuli FSM (Final State Module) lähettää resurssien hallinta- ja varauslohkoston RAM ja sitä kautta edelleen istuntojen hallinta- ja ajoituslohkoston ASM takaisinohjaussanoman (Out sanoma) ilmoittaen vastaanotetusta sanomasta. Odotuslohkoston IB suoritus päättyy, kun lohkosäiliön suorituslogiikka siirtyy jonkin sovelluslohkoston alkuun odotustilamoduulissa ISM vastaanotetun seuraavan ohjauskäskysanoman (Do sa-

## 27

- noma) perusteella. Tällöin suorituslogiikka siirtyy odotuslohkon IB lopussa olevasta päätösmoduulista FSM (Final State Module) haarautumismoduulin BSM kautta seuraavaksi suoritettavan sovelluslohkon alkuun. Istuntojen hallinta- ja ajoituslohko ASM voi erityisellä parametrilla
- 5 ohjauskäskysanomassa määrätä, että suorituslogiikan tulee siirtyä odotuslohkosta päätösmoduulin FSM jälkeen sovelluslohkosäiliön ulosmenomoduuliin TSM (Terminal State Module) lohkosäiliön suorituksen lopettamiseksi ja säiliön vapauttamiseksi. Ulosmenomoduulin TSM jälkeen sovelluslohkosäiliö on istuntojen hallinta- ja ajoituslohkon
- 10 ASM kannalta vapaasti valittavissa minkä tahansa siinä olevan sovelluslohkon suorittamiseen minkä tahansa sovellusistunnon puitteissa. Silloin kun odotuslohkossa IB vastaanotettu sanoma ei aiheuta suorituslogiikan siirtymistä haarautumismoduuliin BSM tai ulosmenomoduuliin TSM, suorituslogiikka aina palaa odotuslohkon päätösmoduulista
- 15 FSM odotuslohkon alussa olevaan odotustilamoduuliin ISM jatkamaan odottamista.

- Tilanteissa, jossa suorittimen 2 kuormitus on suhteellisen pieni, voi istuntojen hallinta- ja ajoituslohko ASM säätää suorittimen 2 tehonkulutusta pienemmäksi esim. kellotaajuutta pienentämällä. Toisaalta tehonkulutuksen säätö voidaan suorittaa joissakin suorittimissa siten, että ainakin osa sellaisista suorittimen toiminnallisista lohkoista, joita ei juuri
- 20 sillä hetkellä tarvita, asetetaan tehonsäästötilaan.

- 25 Nyt esillä oleva keksintö voidaan suurelta osin toteuttaa ohjelmallisesti esim. suorittimen 2 ohjelmakäskyinä.

- On selvää, että nyt esillä olevaa keksintöä ei ole rajoitettu ainoastaan edellä esitettyihin suoritusmuotoihin, vaan sitä voidaan muunnella
- 30 oheisten patenttivaatimusten puitteissa.

Patenttivaatimukset:

1. Järjestelmä, joka käsittää välineet (2, 4) sovellusistuntojen suorittamiseksi elektroniikkalaitteessa (1), jossa on yksi tai useampi suoritin (2), ja välineet (2) resurssivarausten sekä olennaisesti samanaikaisesti suoritettavana olevien sovellusistuntojen suorituksen vuorottelemiseksi, **tunnettu** siitä, että suoritettava sovellusistunto käsittää yhden tai useamman sovelluslohkon (AB) yhdessä tai useammassa sovelluslohkosäilytyksessä (AC, AC1, AC2, AC3), ja mainituille sovelluslohkoille (AB) on määritetty suoritusjärjestys, että järjestelmä käsittää resurssityyppikohtaisia resurssivälittäjiä (RH) resurssien varauksiksi sovellusistunnon käyttöön, resurssien hallinta- ja varausvälineet (RAM, RAT, RH, RIT) resurssien varausilanteen tutkimiseksi ja tallentamiseksi, sovellusistuntojen hallinta- ja ajoitusvälineet (ASM) ainakin mainitun varaustilanteen perusteella seuraavaksi suoritusvuoroon tulevan sovellusistunnon valitsemiseksi, suoritusvälineet (2, ASM) valitussa sovellusistunnossa suoritusvuorossa olevan sovelluslohkon (AB) valitsemiseksi ja suorittamiseksi, ja järjestelmään on muodostettu mainittujen resurssivälittäjien, resurssien hallinta- ja varausvälineiden, sovellusistuntojen hallinta- ja ajoitusvälineiden ja suoritusvälineiden välinen yhteyskäytäntö (SCP) suoritusjärjestyksen määrittämiseksi ja tiedonsiirron toteuttamiseksi mainittujen varausvälineiden, resurssien hallinta- ja varausvälineiden, sovellusistuntojen hallinta- ja ajoitusvälineiden ja suoritusvälineiden välillä.

2. Patenttivaatimuksen 1 mukainen järjestelmä, **tunnettu** siitä, että se käsittää välineet (RAM, RAT, RH, RIT) resurssien varaustilanteen ylläpitämiseksi, välineet (ASM, Do) ohjauskäskysanoman lähettämiseksi sovelluslohkolle resurssien varaamista koskevan ohjaustiedon välittämiseksi sovelluslohkolle sovelluslohkon käynnistytyn yhteydessä, ja välineet takaisinohjaussanoman lähettämiseksi sovelluslohkon suorituksen päättymisen yhteydessä sovelluslohkon suorittamien resurssien varausten ja vapautusten ilmoittamiseksi resurssien varaustilanteen pitämiseksi ajantasalla kunkin sovelluslohkon suorituksen päätyttyä.

3. Patenttivaatimuksen 2 mukainen järjestelmä, **tunnettu** siitä, että se käsittää välineet (ASM, AC, SCP, RAM, RAT, RH, RIT) kunkin sovel-  
 luslohkon tarvitsemien resurssien varaamiseksi sovelluksen käyttöön  
 sekä vapauttamiseksi pois käytöstä joko suoraan resurssityyppikohtai-  
 5 silta resurssivälittimiltä tai resurssipyyntöjen jonotuksen mahdollistavilta  
 resurssien hallinta- ja varausvälineiltä lähtötyössä ohjauskäskysano-  
 massa saadun ohjaustiedon perusteella.
4. Patenttivaatimuksen 2 tai 3 mukainen järjestelmä, **tunnettu** siitä,  
 10 että se käsittää välineet (AC, SCP, RIT, RH) sovellusistunnon tekemien  
 resurssivarausten saamiseksi takaisinohjaussanomien avulla istunnon  
 suoritukseen osallistuvien eri sovelluslohkosäiliöiden käyttöön dynaa-  
 misesti tarpeen mukaan.
5. Jonkin patenttivaatimuksen 1—4 mukainen järjestelmä, **tunnettu**  
 siitä, että järjestelmä käsittää käyttöjärjestelmän, jossa on ajoitusto-  
 15 minnitoja, ja että sovellusistuntojen hallinta- ja ajoitusvälineiden, sovel-  
 luslohkosäiliöiden ja resurssien hallinta- ja varausvälineiden sekä re-  
 surssivälittimien yhdessä suorittamaa resurssien varaamisen ja va-  
 20 pauttamisen ajoitusta ja muuta hallintaa varten on muodostettu sovel-  
 lusriippumattomien ohjaussanomien ja niiden käytösääntöjen muo-  
 dostama istunnonohjausprotokolla, joka toimiessaan on järjestetty to-  
 teuttamaan sovellusistuntojen hallinta- ja ajoitusvälineiden, sovellus-  
 25 lohkosäiliöiden, resurssien hallinta- ja varausvälineiden sekä resurssi-  
 välittimien suorituksen ajoituksen ohjaus käyttöjärjestelmän ajoitusto-  
 imintojen sekä sovellusistuntojen hallinta- ja ajoitusvälineille, sovellus-  
 lohkosäiliöille, resurssien hallinta- ja varausvälineille sekä resurssivälit-  
 timille määritettyjen prioriteettien avulla.
6. Jonkin patenttivaatimuksen 1—5 mukainen järjestelmä, **tunnettu**  
 siitä, että se käsittää resurssivaraustaulun (RIT) kutakin resurssiväli-  
 30 timiä kohti resurssien varaustilanteen välittämiseksi mainituille hallinta-  
 ja varausvälineille (RAM), ja että resurssien hallinta- ja varausvälinei-  
 den ja resurssivälittimien (RH) keskinäinen suorituserjestys on mää-  
 35 rätty siten, että olennaisesti välittömästi resurssivälittimien (RH) suori-  
 tuksen jälkeen ovat suorituvuorossa resurssien hallinta- ja varausväli-

## 30

neet (RAM), jolloin resurssien varaustilanne on tapahtuneiden muutosten osalta yksikäsitteinen resurssivaraustauluissa (RIT).

- 5 7. Patenttivaatimuksen 6 mukainen järjestelmä, **tunnettu** siitä, että resurssien hallinta- ja varausvälineiden ja sovellusistuntojen hallinta- ja ajoitusvälineiden keskinäisen suorituserjestys on määrätty siten, että olennaisesti välittömästi resurssien hallinta- ja varausvälineiden suorituksen jälkeen ovat suoritusvuorossa sovellusistuntojen hallinta- ja ajoitusvälineet (ASM), jolloin resurssien varaustilanne on tapahtuneiden muutosten osalta yksikäsitteinen ja erityyppisten resurssivarausten käytön synkronointiin pyrkivä ohjaustieto on mahdollista muodostaa sovellusistuntojen hallinta- ja ajoitusvälineissä sen lähettämiin ohjauskäskysanomiin.
- 10 8. Jonkin patenttivaatimuksen 1—7 mukainen järjestelmä, **tunnettu** siitä, että kunkin sovelluslohkon loppuun on sijoitettu yksi suorituksen pysäytysmoduuli (SM), sovelluslohkon sisältävän sovelluslohosäiliöön on sijoitettu odotustilamoduuli (ISM), ja että sovelluslohkon sisältävän sovelluslohosäiliön suoritus on järjestetty lähettämään takaisinohjaussanoma pysäytysmoduulissa (SM) ja pysähtymään odotustilamoduulissa (ISM) ja odottamaan istuntojen hallinta- ja ajoituslohoselta ohjauskäskysanomaa, jolloin sovellusistunnon suoritus on kyseisessä lohosäiliössä keskeytetty.
- 15 9. Patenttivaatimuksen 8 mukainen järjestelmä, **tunnettu** siitä, että sovellusistuntojen hallinta- ja ajoitusvälineet on järjestetty suorittamaan resurssien varauksilanteen ja suoritettavien istuntojen ajoituksen analysointia yhtä tai useampaa resurssia koskevan ylikuormitusilanteen havaitsemiseksi sekä hallitsemiseksi korvaamalla tarpeen mukaan sovellusistuntoja vähemmän resurssia tarvitsevilla sovellusistunnoilla, tai viivästyttämällä tarvittaessa sovellusistunnoille lähetettävien ohjauskäskysanomioiden lähettämistä, jolloin käynnissä oleva sovellusistunto on järjestetty keskeytettäväksi tilapäisesti tai uuden sovellusistunnon käynnistystä on järjestetty viivästettäväksi.
- 20 10. Jonkin patenttivaatimuksen 1—9 mukainen järjestelmä, **tunnettu** siitä, että mainituista sovellusistunnon sovelluslohosista on muodos-
- 25
- 30
- 35



## 31

tettu yksi tai useampi sovelluslohkosäiliö, että yhdessä sovelluslohkosäiliössä olevia sovelluslohoja on järjestetty suoritettavaksi ajallisesti eriaikaisesti, ja että mikäli sovellusistunnossa on olennaisesti samanaikaisesti suoritettavaksi tarkoitettuja sovelluslohoja, on ne sijoitettu eri sovelluslohkosäiliöihin.

11. Patenttivaatimuksen 10 mukainen järjestelmä, tunnettu siitä, että sovellusten muodostamiseksi järjestelmässä suoritettavaksi on kuhunkin sovelluslohkosäiliöön muodostettu liitäntärajapinta kohtiin, joissa sovelluslohkon tai sovelluslohkosäiliön suoritus voi pysähtyä ja suoritusvuoro voi vaihtua, jonka liitäntärajapinnan avulla sovelluslohkosäiliössä on järjestetty suoritettavaksi istunnonohjaussanomien lähetys ja vastaanotto ilman, että sovellusta muodostettaessa tarvitsee käsitellä istunnonohjausprotokollan sanomia.

12. Jonkin patenttivaatimuksen 1—11 mukainen järjestelmä, tunnettu siitä, että resurssivälittimissä (RH) on liityntärajapinta tietojen välittämiseksi resurssivälittimen ja järjestelmän välillä, joka liityntärajapinta on olennaisesti riippumaton sovellusistunnosta ja resurssin tyypistä.

13. Jonkin patenttivaatimuksen 1—12 mukainen järjestelmä, tunnettu siitä, että se käsittää kunkin resurssivälittimen (RH) käyttöön järjestetyn resurssivaraustaulun (RIT), ja että resurssivälittimet on muodostettu välitilattomiksi, jolloin resurssivälittimet on järjestetty tallentamaan resurssivarausten tietoja koskevat muutokset resurssivaraustauluun (RIT).

14. Jonkin patenttivaatimuksen 1—13 mukainen järjestelmä, tunnettu siitä, että sovellusistuntojen hallinta- ja ajoitusvälineet (ASM) käsittävät istuntohistoriataulun (SHT), ja resurssien hallinta- ja varausvälineet käsittävät resurssienjakotaulun (RAT), että sovellusistuntojen hallinta- ja ajoitusvälineet (ASM) ja resurssien hallinta- ja varausvälineet on muodostettu välitilattomiksi, jolloin istuntojen tietoja koskevat muutokset on järjestelley tallennettavaksi mainittuun istuntohistoriatauluun (SHT), ja resurssivarausten tietoja koskevat muutokset on järjestetty tallennettavaksi mainittuun resurssienjakotauluun (RAT).

15. Jonkin patenttivaatimuksen 1—14 mukainen järjestelmä, **tunnettu** siitä, että se käsittää välineet (ASM) suorittimon kuormitustilanteen selvittämiseksi ja suorittimen tehonkulutuksen säätämiseksi kuormitustilanteen perusteella sekä sovellusistuntojen suorittamien tehtävien ajoittamisella.

16. Menetelmä sovellusistuntojen suorittamiseksi elektroniikkalaitteessa (1), jossa toimintaa ohjataan yhdellä tai useammalla suorittimella (2), ja jossa menetelmässä resurssivaroja sekä olennaisesti samanaikaisesti suoritettavana olevien sovellusistuntojen suoritusta vuorotellaan, **tunnettu** siitä, että suoritettava sovellusistunto käsittää yhden tai useamman sovelluslohkon (AB) yhdessä tai useammassa sovelluslohosäiliössä (AC, AC1, AC2, AC3), ja mainituille sovelluslokoille (AB) määritetään suoritusjärjestys, että menetelmässä suoritetaan ainakin seuraavia vaiheita:

- resurssien hallinta- ja varausvaihe resurssien varaamiseksi sovellusistunnon käyttöön,
- tutkimis- ja tallennusvaihe resurssien varaustilanteen tutkimiseksi ja tallentamiseksi,
- valintavaihe seuraavaksi suoritusvuoroon tulevan sovellusistunnon valitsemiseksi ainakin mainitun varaustilanteen perusteella,
- suoritusvaihe valitussa sovellusistunnossa suoritusvuorossa olevan sovelluslohkon (AB) valitsemiseksi ja suorittamiseksi.

jolloin menetelmässä käytetään mainittujen resurssien hallinta- ja varausvaiheen, tutkimis- ja tallennusvaiheen, valintavaiheen, ja suoritusvaiheen välistä yhteyskäytäntöä (SCP) suoritusjärjestyksen määrittämiseksi ja tarvittaessa tiedon siirtämiseksi mainittujen resurssien hallinta- ja varausvaiheen, tutkimis- ja tallennusvaiheen, valintavaiheen, ja suoritusvaiheen välillä.

17. Patenttivaatimuksen 16 mukainen menetelmä, **tunnettu** siitä, että ylläpidetään resurssien varaustilannetta, ja lähetetään ohjauskäskeä noma sovelluslokoille resurssien varaamista koskevan ohjaustiedon välittämiseksi sovelluslokoille sovelluslohkon käynnistykseen yhteydessä, ja lähetetään takaisinohjaussanoma sovelluslokoista sovelluslokojen suorittamien resurssien varausten ja vapautusten ilmoittamiseksi

resurssien varaustilanteen pitämiseksi ajantasalla kunkin sovelluslohkon suorituksen päätyttyä.

5 18. Patenttivaatimuksen 17 mukainen menetelmä, **tunnettu** siitä, että kunkin sovelluslohkon tarvitsemia resursseja varataan sovelluksen käyttöön sekä vapautetaan pois käytöstä joko suoraan resurssityyppi-kohtaisilta resurssivälittimiltä tai resurssipyyntöjen jonotuksen mahdollistavilta resurssien hallinta- ja varausvälineiltä lähetetyssä ohjauskysymyksessä saadun ohjaustiedon perusteella.

10

19. Patenttivaatimuksen 17 tai 18 mukainen menetelmä, **tunnettu** siitä, että takaisinohjaussanomia käytetään sovellusistunnon tekemien resurssivarausten ottamiseksi istunnon suoritukseen osallistuvien eri sovelluslohkosaalioiden käyttöön dynaamisesti tarpeen mukaan.

15

20. Jonkin patenttivaatimuksen 16—19 mukainen menetelmä, **tunnettu** siitä, että menetelmässä käytetään käyttöjärjestelmää, jossa on ajoitustoimintoja, ja että suoritusvaiheen, sovelluslohkosaalioiden ja resurssien hallinta- ja varausvaiheen sekä resurssivälittimien yhdessä suorittamaa resurssien varaamisen ja vapauttamisen ajoitusta ja muuta hallintaa varten käytetään sovellusriippumattomien ohjaussanomien ja niiden käyttösääntöjen muodostamaa istunnonohjausprotokollaa, joka toimicssaan totouttaa suoritusvaiheen, sovelluslohkosaalioiden, resurssien hallinta- ja varausvaiheen sekä resurssivälittimien suorituksen ajoituksen ohjausta käyttöjärjestelmän ajoitustoimintojen sekä suoritusvaiheen, sovelluslohkosaalioiden, resurssien hallinta- ja varausvaiheen sekä resurssivälittimille määritettyjen prioriteettien avulla.

20

25

30

21. Jonkin patenttivaatimuksen 16—20 mukainen menetelmä, **tunnettu** siitä, että menetelmässä käytetään resurssivälittinkohtaista resurssivaraustaulua (RIT) kunkin resurssin varaustilanteen välittämiseksi mainittuun resurssien hallinta- ja varausvaiheeseen, ja että resurssien hallinta- ja varausvaiheen ja resurssivälittimien (RH) keskinäinen suorituserjestys on määrätty siten, että olennaisesti välittömästi resurssivälittimien (RH) suorituksen jälkeen on suoritussuorossa resurssien hallinta- ja varausvaihe, jolloin resurssien varaustilanne on

35

tapahtuneiden muutosten osalta yksikäsitteinen resurssivaraustauluis-  
sa (RIT).

5 22. Patenttivaatimuksen 21 mukainen menetelmä, **tunnettu** siitä, että  
resurssien hallinta- ja varausvaiheen ja suoritusvaiheen keskinäinen  
suoritusjärjestys määrätään siten, että olennaisesti välittömästi resurs-  
sien hallinta- ja varausvaiheen suorituksen jälkeen on suoritusvuorossa  
suoritusvaihe, jolloin resurssien varaustilanne on tapahtuneiden muu-  
tosten osalla yksikäsitteinen ja erityyppisten resurssivarausten käytön  
10 synkronointiin pyrkivä ohjaustieto on mahdollista muodostaa suoritus-  
vaiheessa siinä lähettäviin ohjauskäskysanomiiin.

15 23. Jonkin patenttivaatimuksen 16—23 mukainen menetelmä,  
**tunnettu** siitä, että kunkin sovelluslohkon loppuun sijoitetaan yksi suo-  
rituksen pysäytysmoduuli (SM), sovelluslohkon sisältävään sovellus-  
lohkosaäiliöön sijoitetaan odotustilamoduuli (ISM), ja että sovelluslohkon  
sisältävän sovelluslohkosaäiliön suorituksessa lähetetään takaisinoh-  
jaussanoma sovelluslohkosaäiliön pysäytysmoduulissa (SM) ja pysäh-  
dytään odotustilamoduulissa (ISM) ja odotetaan istuntojen hallinta- ja  
20 ajoituslohkolta ohjauskäskysanomaa, jolloin sovellusistunnon suoritus  
on kyseisessä lohkosäiliössä keskeytettyinä.

25 24. Patenttivaatimuksen 23 mukainen menetelmä, **tunnettu** siitä, että  
suoritusvaiheessa suoritetaan resurssien varaustilanteen ja suoritetta-  
vien istuntojen ajoituksen analysointia yhtä tai useampaa resurssia  
koskevan ylikuormitustilanteen havaitsemiseksi sekä hallitsemiseksi  
korvaamalla tarpeen mukaan sovellusistuntoja vähemmän resursseja  
tarvitsevilla sovellusistunnoilla, tai viivästyttämällä tarvittaessa sovellus-  
istunnoille lähetettävien ohjauskäskysanomien lähettämistä, jolloin  
30 käynnissä oleva sovellusistunto keskeytetään tilapäisesti tai uuden so-  
vellusistunnon käynnistystä viivästetään.

35 25. Jonkin patenttivaatimuksen 16—24 mukainen menetelmä,  
**tunnettu** siitä, että mainituista sovellusistunnon sovelluslohkoista  
muodostetaan yksi tai useampi sovelluslohkosaäiliö, että yhdessä so-  
velluslohkosaäiliössä olevia sovelluslohkoja suoritetaan ajallisesti eriai-  
kaisesti, ja että mikäli sovellusistunnossa on olennaisesti samanaikai-

sesti suoritettavaksi tarkoitettuja sovelluslohkoja, sijoitetaan ne eri sovelluslohkosäiliöihin.

5 26. Patenttivaatimuksen 25 mukainen menetelmä, **tunnettu** siitä, että sovellusten muodostamiseksi järjestelmässä suoritettavaksi kuhunkin sovelluslohkosäiliöön muodostetaan liitántärajapinta kohtiin, joissa sovelluslohkon tai sovelluslohkosäiliön suoritus voi pysähtyä ja suoritus-  
10 vuoro voi vaihtua, jonka liitántärajapinnan avulla sovelluslohkosäiliössä suoritetaan istunnonohjaussarjien lähetyks ja vastaanotto ilman, että sovellusta muodostettaessa tarvitsee käsitellä istunnonohjausprotokollan sanomia.

15 27. Jonkin patenttivaatimuksen 16—26 mukainen menetelmä, **tunnettu** siitä, että resurssivälittimissä (RH) käytetään liityntärajapintaa tietojen välittämiseksi resurssivälittimiin ja resurssivälittimiltä, joka liityntärajapinta on olennaisesti riippumaton sovellusistunnosta ja resurs-  
sin tyypistä.

20 28. onkin patenttivaatimuksen 16—27 mukainen menetelmä, **tunnettu** siitä, että kunkin resurssivälittimen (RH) käytössä on resurssivaraustaulu (RIT), ja että resurssivälittimet muodostetaan välitilattomiksi, jolloin resurssivälittimet tallentavat resurssivarauksen tietoja koskevat muutokset resurssivaraustauluun (RIT).

25 29. Jonkin patenttivaatimuksen 16—28 mukainen menetelmä, **tunnettu** siitä, että suoritusvaihetta varten muodostetaan istuntohistoriataulu (SHT), ja resurssien hallinta- ja varausvaihetta varten resurssienjakotaulu (RAT). että suoritusvaihe ja resurssien hallinta- ja varausvaihe muodostetaan välitilattomiksi, jolloin istuntojen tietoja koskevat  
30 muutokset tallennetaan mainittuun istuntohistoriatauluun (SHT), ja resurssivarausten tietoja koskevat muutokset tallennetaan mainittuun resurssienjakotauluun (RAT).

35 30. Jonkin patenttivaatimuksen 16—29 mukainen menetelmä, **tunnettu** siitä, että suorittimen kuormitustilanne selvitetään, ja että suorittimen tehonkulutusta säädetään kuormitustilanteen perusteella sekä sovellusistuntojen suorittamien tehtävien ajoittamisella.

31. Elektroniikkalaito (1), joka käsittää välineet (2, 4) sovellusistuntojen suorittamiseksi, yhden tai useamman suorittimen (2), ja välineet (2) resurssivarausten sekä olennaisesti samanaikaisesti suoritettavana olevien sovellusistuntojen suorituksen vuorottelemiseksi, **tunnettu** siitä, että suoritettava sovellusistunto käsittää yhden tai useamman sovelluslohkon (AB) yhdessä tai useammassa sovelluslohkosäiliössä (AC, AC1, AC2, AC3), ja mainituille sovelluslohkoille (AB) on määritetty suoritusjärjestys, että elektroniikkalaitteeseen (1) käsittää resurssityypikohtaisia resurssivälittäjiä (RH) resurssien varaamiseksi sovellusistunnon käyttöön, resurssien hallinta- ja varausvälineet (RAM, RAT, RII, RIT) resurssien varaus tilanteen tutkimiseksi ja tallentamiseksi, sovellusistuntojen hallinta- ja ajoitusvälineet (ASM) ainakin mainitun varaus tilanteen perusteella seuraavaksi suoritusvuoroon tulevan sovellusistunnon valitsemiseksi, suoritusvälineet (2, ASM) valitussa sovellusistunnossa suoritusvuorossa olevan sovelluslohkon (AB) valitsemiseksi ja suorittamiseksi, ja elektroniikkalaitteeseen (1) on muodostettu välineet (SCP) mainittujen resurssivälittäjien, resurssien hallinta- ja varausvälineiden, sovellusistuntojen hallinta- ja ajoitusvälineiden ja suoritusvälineiden välisen yhteyksikäytännön (SCP) suoritusjärjestyksen määrittämiseksi ja tiedonsiirron toteuttamiseksi mainittujen varausvälineiden, resurssien hallinta- ja varausvälineiden, sovellusistuntojen hallinta- ja ajoitusvälineiden ja suoritusvälineiden välillä.
- 5
- 10
- 15
- 20
- 25
32. Patenttivaatimuksen 31 mukainen elektroniikkalaitte (1), **tunnettu** siitä, että se on langaton viestintälaitte.

## (57) Tiivistelmä

Nyt esillä oleva keksintö kohdistuu järjestelmään, joka käsittää välineet (2, 4) sovellusistuntojen suorituksen sekä tarvittavien resurssivarausten ajoittamiseksi elektroniikkalaitteessa (1), jossa on yksi tai useampi suoritin (2). Järjestelmä käsittää välineet (2) resurssivarausten sekä olennaisesti samanaikaisesti suoritettavana olevien sovellusistuntojen suorituksen vuorollelemiseksi. Suoritettavasta sovellusistunnosta on muodostettu yksi tai useampi sovelluslohko (AB) yhteen tai useampaan sovelluslohkosäiliöön (AC, AC1, AC2, AC3), ja mainituille sovelluslohkoille (AB) on määritelty suoritusjärjestys. Järjestelmä käsittää resurssityyppikohtaisia resurssivälittäjiä (RH) resurssien varaamiseksi sovellusistunnon käyttöön, resurssien hallinta- ja varausvälineet (RAM, RAT, RH, RIT) resurssien varaustilanteen tutkimiseksi ja tallentamiseksi, sovellusistuntojen hallinta- ja ajoitusvälineet (ASM) ainakin mainitun varaustilanteen perusteella seuraavaksi suoritusvuoroon tulevan sovellusistunnon valitsemiseksi, suoritusvälineet (2, ASM) valitussa sovellusistunnossa suoritusvuorossa olevan sovelluslohkon (AB) valitsemiseksi ja suorittamiseksi. Järjestelmään on muodostettu mainittujen resurssivälittäjien, resurssien hallinta ja varausvälineiden, sovellusistuntojen hallinta- ja ajoitusvälineiden ja suoritusvälineiden välinen yhteyskäytäntö (SCP) suoritusjärjestyksen määrittämiseksi ja tiedonsiirron toteuttamiseksi mainittujen varausvälineiden, resurssien hallinta- ja varausvälineiden, sovellusistuntojen hallinta- ja ajoitusvälineiden ja suoritusvälineiden välillä. Keksintö kohdistuu lisäksi menetelmään sovellusistuntojen suorittamiseksi elektroniikkalaitteessa (1) sekä elektroniikkalaitteeseen (1).

Fig. 4

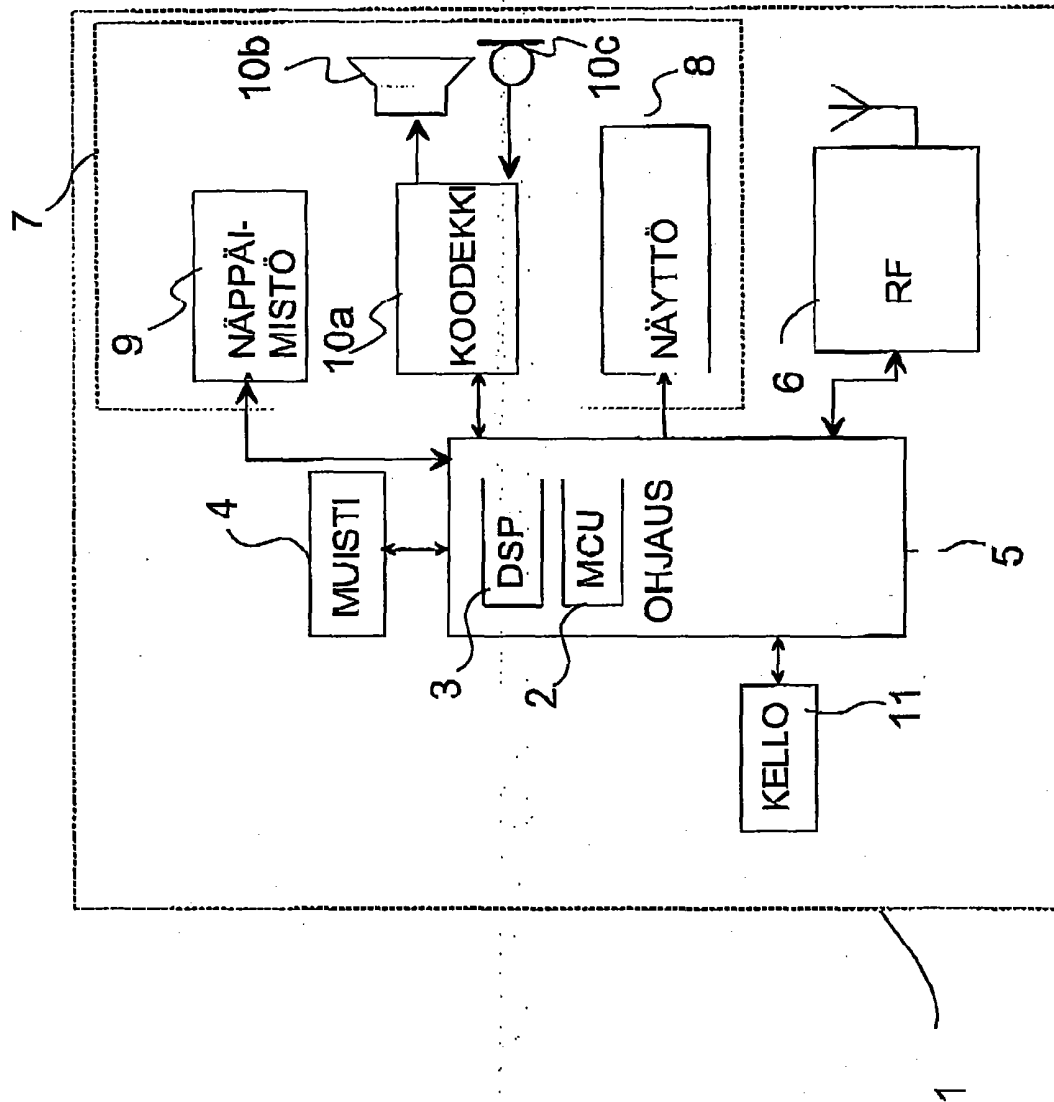
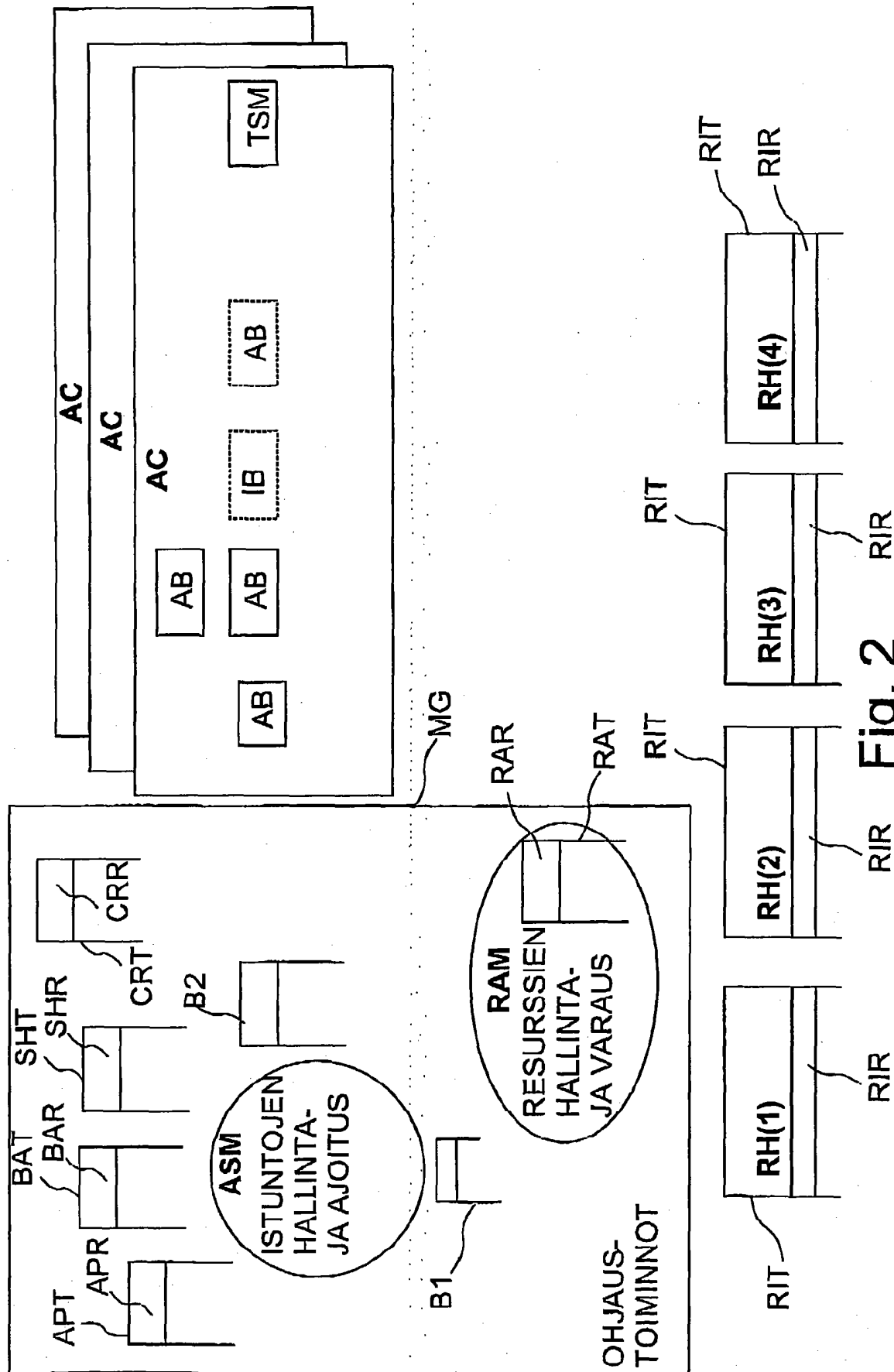


Fig. 1







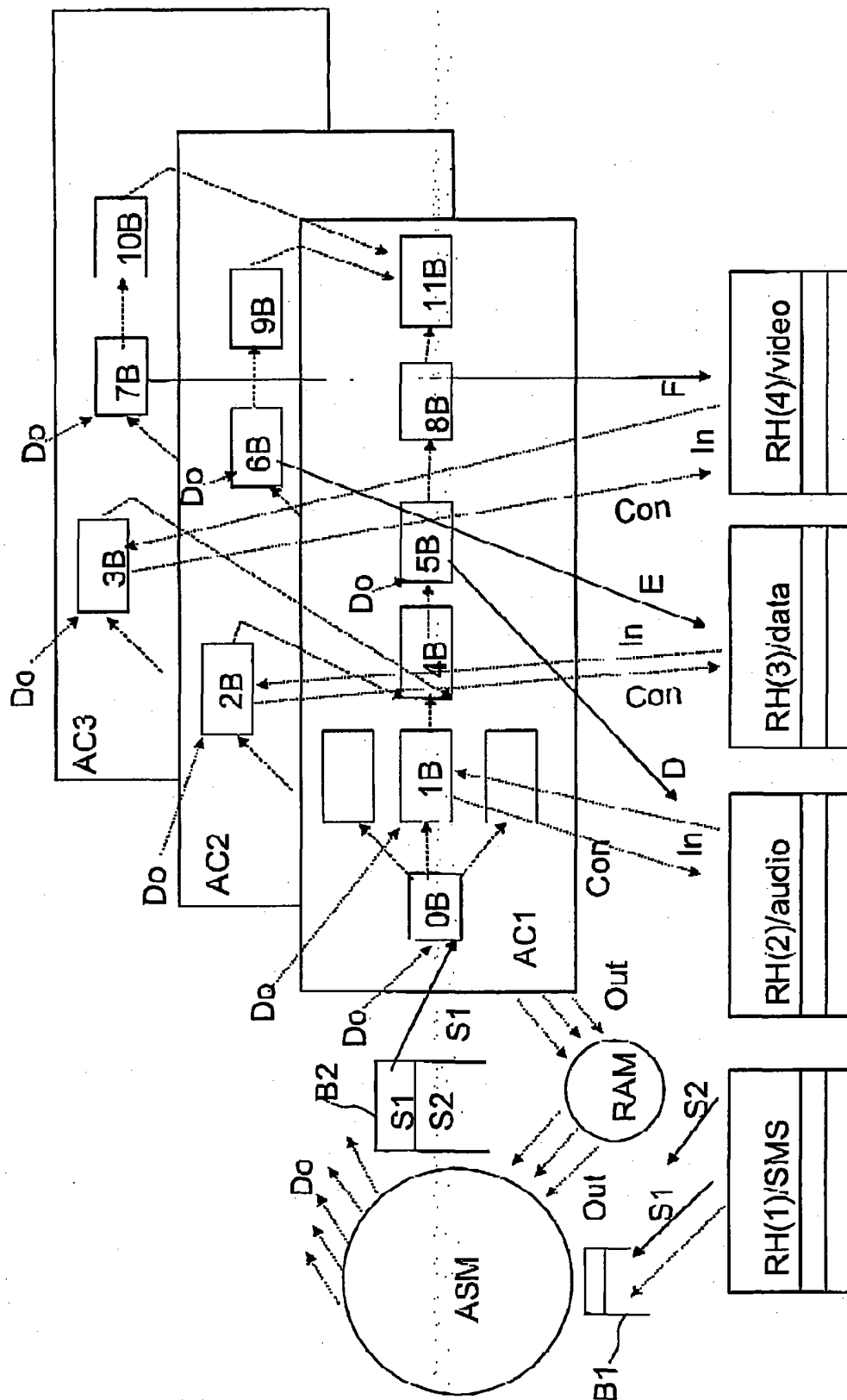


Fig. 4